```
{
 "cells": [
  {
   "cell_type": "code",
   "execution_count": 1,
   "id": "0faff956",
   "metadata": {
    "execution": {
     "iopub.execute_input": "2021-12-21T16:51:18.859702Z",
     "iopub.status.busy": "2021-12-21T16:51:18.858453Z",
     "iopub.status.idle": "2021-12-21T16:51:26.306435Z",
     "shell.execute_reply": "2021-12-21T16:51:26.305604Z",
     "shell.execute_reply.started": "2021-12-21T16:21:45.336780Z"
    },
    "id": "tPXrnm0Dn2ek",
    "papermill": {
     "duration": 7.479767,
     "end_time": "2021-12-21T16:51:26.306610",
     "exception": false,
     "start_time": "2021-12-21T16:51:18.826843",
     "status": "completed"
    },
    "tags": []
   },
   "outputs": [],
   "source": [
    "import pandas as pd\n",
    "import numpy as np\n",
    "import matplotlib.pyplot as plt\n",
    "%matplotlib inline\n",
    "import seaborn as sns\n",
    "import tensorflow as tf\n",
    "from tensorflow import keras\n",
    "from tensorflow.keras.models import Sequential\n",
    "from tensorflow.keras.layers import Dense,Flatten,Dropout\n",
    "from tensorflow.keras.optimizers import Adam\n",
    "from tensorflow.keras.models import load_model\n",
    "#from keras.utils import to_categorical\n",
    "#importing models\n",
    "from sklearn.model_selection import train_test_split\n",
    "from sklearn.preprocessing import LabelEncoder, MinMaxScaler\n",
    "import time\n",
```

```
   "import warnings\n",
      "from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score, classification_report, confusion_matrix"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 2,
   "id": "8674b4f2",
   "metadata": {
    "execution": {
     "iopub.execute_input": "2021-12-21T16:51:26.369062Z",
     "iopub.status.busy": "2021-12-21T16:51:26.367918Z",
     "iopub.status.idle": "2021-12-21T16:51:26.370692Z",
     "shell.execute_reply": "2021-12-21T16:51:26.370116Z",
     "shell.execute_reply.started": "2021-12-21T16:22:00.467881Z"
    },
    "papermill": {
     "duration": 0.036674,
     "end_time": "2021-12-21T16:51:26.370869",
     "exception": false,
     "start_time": "2021-12-21T16:51:26.334195",
     "status": "completed"
    },
    "tags": []
   },
   "outputs": [],
   "source": [
    "from sklearn.preprocessing import StandardScaler"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 3,
   "id": "11002db1",
   "metadata": {
    "execution": {
     "iopub.execute_input": "2021-12-21T16:51:26.433348Z",
     "iopub.status.busy": "2021-12-21T16:51:26.432606Z",
     "iopub.status.idle": "2021-12-21T16:51:26.447538Z",
     "shell.execute_reply": "2021-12-21T16:51:26.448124Z",
     "shell.execute_reply.started": "2021-12-21T16:22:03.446706Z"
```

```
  },
  "papermill": {
   "duration": 0.049466,
   "end_time": "2021-12-21T16:51:26.448369",
   "exception": false,
   "start_time": "2021-12-21T16:51:26.398903",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "/kaggle/input/churn-modellingcsv/Churn_Modelling.csv\n"
   ]
  }
 ],
 "source": [
  "import os\n",
  "for dirname, _, filenames in os.walk('/kaggle/input'):\n",
  "    for filename in filenames:\n",
  "        print(os.path.join(dirname, filename))"
 ]
},
{
 "cell_type": "code",
 "execution_count": 4,
 "id": "ece44297",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:26.509075Z",
   "iopub.status.busy": "2021-12-21T16:51:26.507831Z",
   "iopub.status.idle": "2021-12-21T16:51:26.568633Z",
   "shell.execute_reply": "2021-12-21T16:51:26.567940Z",
   "shell.execute_reply.started": "2021-12-21T16:22:18.140727Z"
  },
  "id": "W6My8YVOpFYl",
  "papermill": {
   "duration": 0.091685,
   "end_time": "2021-12-21T16:51:26.568839",
```

```
   "exception": false,
   "start_time": "2021-12-21T16:51:26.477154",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "df = pd.read_csv('/kaggle/input/churn-modellingcsv/Churn_Modelling.csv')"
 ]
},
{
 "cell_type": "code",
 "execution_count": 5,
 "id": "b994024f",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:26.645751Z",
   "iopub.status.busy": "2021-12-21T16:51:26.644842Z",
   "iopub.status.idle": "2021-12-21T16:51:26.671756Z",
   "shell.execute_reply": "2021-12-21T16:51:26.672337Z",
   "shell.execute_reply.started": "2021-12-21T16:22:19.827533Z"
  },
  "id": "J6epjyw1pFgc",
  "papermill": {
   "duration": 0.075692,
   "end_time": "2021-12-21T16:51:26.672546",
   "exception": false,
   "start_time": "2021-12-21T16:51:26.596854",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/html": [
     "<div>\n",
     "<style scoped>\n",
     "    .dataframe tbody tr th:only-of-type {\n",
     "        vertical-align: middle;\n",
     "    }\n",
```

```
"\n",
"    .dataframe tbody tr th {\n",
"        vertical-align: top;\n",
"    }\n",
"\n",
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>RowNumber</th>\n",
"      <th>CustomerId</th>\n",
"      <th>Surname</th>\n",
"      <th>CreditScore</th>\n",
"      <th>Geography</th>\n",
"      <th>Gender</th>\n",
"      <th>Age</th>\n",
"      <th>Tenure</th>\n",
"      <th>Balance</th>\n",
"      <th>NumOfProducts</th>\n",
"      <th>HasCrCard</th>\n",
"      <th>IsActiveMember</th>\n",
"      <th>EstimatedSalary</th>\n",
"      <th>Exited</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>1</td>\n",
"      <td>15634602</td>\n",
"      <td>Hargrave</td>\n",
"      <td>619</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>42</td>\n",
"      <td>2</td>\n",
"      <td>0.00</td>\n",
"      <td>1</td>\n",
```

```
"    <td>1</td>\n",
"    <td>1</td>\n",
"    <td>101348.88</td>\n",
"    <td>1</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>1</th>\n",
"    <td>2</td>\n",
"    <td>15647311</td>\n",
"    <td>Hill</td>\n",
"    <td>608</td>\n",
"    <td>Spain</td>\n",
"    <td>Female</td>\n",
"    <td>41</td>\n",
"    <td>1</td>\n",
"    <td>83807.86</td>\n",
"    <td>1</td>\n",
"    <td>0</td>\n",
"    <td>1</td>\n",
"    <td>112542.58</td>\n",
"    <td>0</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>2</th>\n",
"    <td>3</td>\n",
"    <td>15619304</td>\n",
"    <td>Onio</td>\n",
"    <td>502</td>\n",
"    <td>France</td>\n",
"    <td>Female</td>\n",
"    <td>42</td>\n",
"    <td>8</td>\n",
"    <td>159660.80</td>\n",
"    <td>3</td>\n",
"    <td>1</td>\n",
"    <td>0</td>\n",
"    <td>113931.57</td>\n",
"    <td>1</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>3</th>\n",
"    <td>4</td>\n",
```

```
"     <td>15701354</td>\n",
"     <td>Boni</td>\n",
"     <td>699</td>\n",
"     <td>France</td>\n",
"     <td>Female</td>\n",
"     <td>39</td>\n",
"     <td>1</td>\n",
"     <td>0.00</td>\n",
"     <td>2</td>\n",
"     <td>0</td>\n",
"     <td>0</td>\n",
"     <td>93826.63</td>\n",
"     <td>0</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>4</th>\n",
"     <td>5</td>\n",
"     <td>15737888</td>\n",
"     <td>Mitchell</td>\n",
"     <td>850</td>\n",
"     <td>Spain</td>\n",
"     <td>Female</td>\n",
"     <td>43</td>\n",
"     <td>2</td>\n",
"     <td>125510.82</td>\n",
"     <td>1</td>\n",
"     <td>1</td>\n",
"     <td>1</td>\n",
"     <td>79084.10</td>\n",
"     <td>0</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>...</th>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
```

```
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"     <td>...</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>9995</th>\n",
"     <td>9996</td>\n",
"     <td>15606229</td>\n",
"     <td>Obijiaku</td>\n",
"     <td>771</td>\n",
"     <td>France</td>\n",
"     <td>Male</td>\n",
"     <td>39</td>\n",
"     <td>5</td>\n",
"     <td>0.00</td>\n",
"     <td>2</td>\n",
"     <td>1</td>\n",
"     <td>0</td>\n",
"     <td>96270.64</td>\n",
"     <td>0</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>9996</th>\n",
"     <td>9997</td>\n",
"     <td>15569892</td>\n",
"     <td>Johnstone</td>\n",
"     <td>516</td>\n",
"     <td>France</td>\n",
"     <td>Male</td>\n",
"     <td>35</td>\n",
"     <td>10</td>\n",
"     <td>57369.61</td>\n",
"     <td>1</td>\n",
"     <td>1</td>\n",
"     <td>1</td>\n",
"     <td>101699.77</td>\n",
"     <td>0</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>9997</th>\n",
```

```
"      <td>9998</td>\n",
"      <td>15584532</td>\n",
"      <td>Liu</td>\n",
"      <td>709</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>36</td>\n",
"      <td>7</td>\n",
"      <td>0.00</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>42085.58</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>9998</th>\n",
"      <td>9999</td>\n",
"      <td>15682355</td>\n",
"      <td>Sabbatini</td>\n",
"      <td>772</td>\n",
"      <td>Germany</td>\n",
"      <td>Male</td>\n",
"      <td>42</td>\n",
"      <td>3</td>\n",
"      <td>75075.31</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>92888.52</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>9999</th>\n",
"      <td>10000</td>\n",
"      <td>15628319</td>\n",
"      <td>Walker</td>\n",
"      <td>792</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>28</td>\n",
"      <td>4</td>\n",
```

```
"      <td>130142.79</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>38190.78</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"<p>10000 rows × 14 columns</p>\n",
"</div>"
],
"text/plain": [
"      RowNumber  CustomerId   Surname  CreditScore Geography  Gender  Age  \\\n",
"0           1    15634602  Hargrave          619    France  Female   42   \n",
"1           2    15647311      Hill          608     Spain  Female   41   \n",
"2           3    15619304      Onio          502    France  Female   42   \n",
"3           4    15701354      Boni          699    France  Female   39   \n",
"4           5    15737888  Mitchell          850     Spain  Female   43   \n",
"...       ...         ...       ...          ...       ...     ...  ...   \n",
"9995     9996    15606229   Obijiaku          771    France    Male   39   \n",
"9996     9997    15569892  Johnstone          516    France    Male   35   \n",
"9997     9998    15584532        Liu          709    France  Female   36   \n",
"9998     9999    15682355  Sabbatini          772   Germany    Male   42   \n",
"9999    10000    15628319     Walker          792    France  Female   28   \n",
"\n",
"      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \\\n",
"0          2       0.00              1          1               1   \n",
"1          1   83807.86              1          0               1   \n",
"2          8  159660.80              3          1               0   \n",
"3          1       0.00              2          0               0   \n",
"4          2  125510.82              1          1               1   \n",
"...      ...        ...            ...        ...             ...   \n",
"9995       5       0.00              2          1               0   \n",
"9996      10   57369.61              1          1               1   \n",
"9997       7       0.00              1          0               1   \n",
"9998       3   75075.31              2          1               0   \n",
"9999       4  130142.79              1          1               0   \n",
"\n",
"      EstimatedSalary  Exited  \n",
"0           101348.88       1  \n",
"1           112542.58       0  \n",
```

```
    "2          113931.57       1  \n",
    "3           93826.63       0  \n",
    "4           79084.10       0  \n",
    "...              ...     ...  \n",
    "9995        96270.64       0  \n",
    "9996       101699.77       0  \n",
    "9997        42085.58       1  \n",
    "9998        92888.52       1  \n",
    "9999        38190.78       0  \n",
    "\n",
    "[10000 rows x 14 columns]"
   ]
  },
  "execution_count": 5,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "df"
]
},
{
"cell_type": "code",
"execution_count": 6,
"id": "9253fe0a",
"metadata": {
 "execution": {
  "iopub.execute_input": "2021-12-21T16:51:26.735356Z",
  "iopub.status.busy": "2021-12-21T16:51:26.734599Z",
  "iopub.status.idle": "2021-12-21T16:51:26.746047Z",
  "shell.execute_reply": "2021-12-21T16:51:26.745490Z",
  "shell.execute_reply.started": "2021-12-21T16:22:20.155393Z"
 },
 "id": "spYXA5FApFpd",
 "papermill": {
  "duration": 0.044519,
  "end_time": "2021-12-21T16:51:26.746221",
  "exception": false,
  "start_time": "2021-12-21T16:51:26.701702",
  "status": "completed"
 },
```

```
 "tags": []
},
"outputs": [
 {
  "data": {
   "text/plain": [
    "RowNumber          0\n",
    "CustomerId         0\n",
    "Surname            0\n",
    "CreditScore        0\n",
    "Geography          0\n",
    "Gender             0\n",
    "Age                0\n",
    "Tenure             0\n",
    "Balance            0\n",
    "NumOfProducts      0\n",
    "HasCrCard          0\n",
    "IsActiveMember     0\n",
    "EstimatedSalary    0\n",
    "Exited             0\n",
    "dtype: int64"
   ]
  },
  "execution_count": 6,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "df.isnull().sum()"
]
},
{
"cell_type": "code",
"execution_count": 7,
"id": "a77ad880",
"metadata": {
 "execution": {
  "iopub.execute_input": "2021-12-21T16:51:26.815145Z",
  "iopub.status.busy": "2021-12-21T16:51:26.814458Z",
  "iopub.status.idle": "2021-12-21T16:51:26.834743Z",
  "shell.execute_reply": "2021-12-21T16:51:26.834175Z",
```

```
  "shell.execute_reply.started": "2021-12-21T16:22:20.475105Z"
 },
 "id": "0guxKKH0pFnv",
 "papermill": {
  "duration": 0.059489,
  "end_time": "2021-12-21T16:51:26.834906",
  "exception": false,
  "start_time": "2021-12-21T16:51:26.775417",
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "name": "stdout",
  "output_type": "stream",
  "text": [
   "<class 'pandas.core.frame.DataFrame'>\n",
   "RangeIndex: 10000 entries, 0 to 9999\n",
   "Data columns (total 14 columns):\n",
   " #   Column           Non-Null Count  Dtype  \n",
   "---  ------           --------------  -----  \n",
   " 0   RowNumber        10000 non-null  int64  \n",
   " 1   CustomerId       10000 non-null  int64  \n",
   " 2   Surname          10000 non-null  object \n",
   " 3   CreditScore      10000 non-null  int64  \n",
   " 4   Geography        10000 non-null  object \n",
   " 5   Gender           10000 non-null  object \n",
   " 6   Age              10000 non-null  int64  \n",
   " 7   Tenure           10000 non-null  int64  \n",
   " 8   Balance          10000 non-null  float64\n",
   " 9   NumOfProducts    10000 non-null  int64  \n",
   " 10  HasCrCard        10000 non-null  int64  \n",
   " 11  IsActiveMember   10000 non-null  int64  \n",
   " 12  EstimatedSalary  10000 non-null  float64\n",
   " 13  Exited           10000 non-null  int64  \n",
   "dtypes: float64(2), int64(9), object(3)\n",
   "memory usage: 1.1+ MB\n"
  ]
 }
],
"source": [
```

```
  "df.info()"
 ]
},
{
 "cell_type": "code",
 "execution_count": 8,
 "id": "1cd8e18d",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:26.901220Z",
   "iopub.status.busy": "2021-12-21T16:51:26.900503Z",
   "iopub.status.idle": "2021-12-21T16:51:26.904837Z",
   "shell.execute_reply": "2021-12-21T16:51:26.904338Z",
   "shell.execute_reply.started": "2021-12-21T16:22:20.757730Z"
  },
  "papermill": {
   "duration": 0.039477,
   "end_time": "2021-12-21T16:51:26.904979",
   "exception": false,
   "start_time": "2021-12-21T16:51:26.865502",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "Index(['RowNumber', 'CustomerId', 'Surname', 'CreditScore', 'Geography',\n",
     "       'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard',\n",
     "       'IsActiveMember', 'EstimatedSalary', 'Exited'],\n",
     "      dtype='object')"
    ]
   },
   "execution_count": 8,
   "metadata": {},
   "output_type": "execute_result"
  }
 ],
 "source": [
  "df.columns"
 ]
```

```
},
{
 "cell_type": "code",
 "execution_count": 9,
 "id": "7139a754",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:26.969846Z",
   "iopub.status.busy": "2021-12-21T16:51:26.968861Z",
   "iopub.status.idle": "2021-12-21T16:51:26.972334Z",
   "shell.execute_reply": "2021-12-21T16:51:26.972800Z",
   "shell.execute_reply.started": "2021-12-21T16:22:21.014201Z"
  },
  "papermill": {
   "duration": 0.038679,
   "end_time": "2021-12-21T16:51:26.972995",
   "exception": false,
   "start_time": "2021-12-21T16:51:26.934316",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "list_drob=['RowNumber','CustomerId','Surname']"
 ]
},
{
 "cell_type": "code",
 "execution_count": 10,
 "id": "4855f44d",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:27.040156Z",
   "iopub.status.busy": "2021-12-21T16:51:27.039360Z",
   "iopub.status.idle": "2021-12-21T16:51:27.042758Z",
   "shell.execute_reply": "2021-12-21T16:51:27.042194Z",
   "shell.execute_reply.started": "2021-12-21T16:22:21.249672Z"
  },
  "papermill": {
   "duration": 0.040751,
   "end_time": "2021-12-21T16:51:27.042899",
```

```
   "exception": false,
   "start_time": "2021-12-21T16:51:27.002148",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "df.drop(list_drob,axis=1,inplace=True)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 11,
 "id": "a12fd064",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:27.117405Z",
   "iopub.status.busy": "2021-12-21T16:51:27.116246Z",
   "iopub.status.idle": "2021-12-21T16:51:27.120943Z",
   "shell.execute_reply": "2021-12-21T16:51:27.120389Z",
   "shell.execute_reply.started": "2021-12-21T16:22:21.487863Z"
  },
  "papermill": {
   "duration": 0.048486,
   "end_time": "2021-12-21T16:51:27.121087",
   "exception": false,
   "start_time": "2021-12-21T16:51:27.072601",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/html": [
     "<div>\n",
     "<style scoped>\n",
     "    .dataframe tbody tr th:only-of-type {\n",
     "        vertical-align: middle;\n",
     "    }\n",
     "\n",
```

```
"    .dataframe tbody tr th {\n",
"        vertical-align: top;\n",
"    }\n",
"\n",
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>CreditScore</th>\n",
"      <th>Geography</th>\n",
"      <th>Gender</th>\n",
"      <th>Age</th>\n",
"      <th>Tenure</th>\n",
"      <th>Balance</th>\n",
"      <th>NumOfProducts</th>\n",
"      <th>HasCrCard</th>\n",
"      <th>IsActiveMember</th>\n",
"      <th>EstimatedSalary</th>\n",
"      <th>Exited</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>619</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>42</td>\n",
"      <td>2</td>\n",
"      <td>0.00</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>101348.88</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
```

```
"      <td>608</td>\n",
"      <td>Spain</td>\n",
"      <td>Female</td>\n",
"      <td>41</td>\n",
"      <td>1</td>\n",
"      <td>83807.86</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>112542.58</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>502</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>42</td>\n",
"      <td>8</td>\n",
"      <td>159660.80</td>\n",
"      <td>3</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>113931.57</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>699</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>39</td>\n",
"      <td>1</td>\n",
"      <td>0.00</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>93826.63</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
```

```
    "      <td>850</td>\n",
    "      <td>Spain</td>\n",
    "      <td>Female</td>\n",
    "      <td>43</td>\n",
    "      <td>2</td>\n",
    "      <td>125510.82</td>\n",
    "      <td>1</td>\n",
    "      <td>1</td>\n",
    "      <td>1</td>\n",
    "      <td>79084.10</td>\n",
    "      <td>0</td>\n",
    "    </tr>\n",
    "  </tbody>\n",
    "</table>\n",
    "</div>"
   ],
   "text/plain": [
    "   CreditScore Geography  Gender  Age  Tenure    Balance  NumOfProducts  \\\n",
    "0          619    France  Female   42       2       0.00              1   \n",
    "1          608     Spain  Female   41       1   83807.86              1   \n",
    "2          502    France  Female   42       8  159660.80              3   \n",
    "3          699    France  Female   39       1       0.00              2   \n",
    "4          850     Spain  Female   43       2  125510.82              1   \n",
    "\n",
    "   HasCrCard  IsActiveMember  EstimatedSalary  Exited  \n",
    "0          1               1        101348.88       1  \n",
    "1          0               1        112542.58       0  \n",
    "2          1               0        113931.57       1  \n",
    "3          0               0         93826.63       0  \n",
    "4          1               1         79084.10       0  "
   ]
  },
  "execution_count": 11,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "df.head()"
]
},
{
```

```
 "cell_type": "code",
 "execution_count": 12,
 "id": "e058eec5",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:27.191614Z",
   "iopub.status.busy": "2021-12-21T16:51:27.187604Z",
   "iopub.status.idle": "2021-12-21T16:51:27.200921Z",
   "shell.execute_reply": "2021-12-21T16:51:27.200292Z",
   "shell.execute_reply.started": "2021-12-21T16:22:21.710619Z"
  },
  "papermill": {
   "duration": 0.04998,
   "end_time": "2021-12-21T16:51:27.201068",
   "exception": false,
   "start_time": "2021-12-21T16:51:27.151088",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "df=pd.get_dummies(df,columns=['Geography','Gender'])"
 ]
},
{
 "cell_type": "code",
 "execution_count": 13,
 "id": "39fa2827",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:27.276656Z",
   "iopub.status.busy": "2021-12-21T16:51:27.275906Z",
   "iopub.status.idle": "2021-12-21T16:51:27.278812Z",
   "shell.execute_reply": "2021-12-21T16:51:27.279326Z",
   "shell.execute_reply.started": "2021-12-21T16:22:21.917300Z"
  },
  "papermill": {
   "duration": 0.048604,
   "end_time": "2021-12-21T16:51:27.279502",
   "exception": false,
   "start_time": "2021-12-21T16:51:27.230898",
```

```
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "data": {
   "text/html": [
    "<div>\n",
    "<style scoped>\n",
    "    .dataframe tbody tr th:only-of-type {\n",
    "        vertical-align: middle;\n",
    "    }\n",
    "\n",
    "    .dataframe tbody tr th {\n",
    "        vertical-align: top;\n",
    "    }\n",
    "\n",
    "    .dataframe thead th {\n",
    "        text-align: right;\n",
    "    }\n",
    "</style>\n",
    "<table border=\"1\" class=\"dataframe\">\n",
    "  <thead>\n",
    "    <tr style=\"text-align: right;\">\n",
    "      <th></th>\n",
    "      <th>CreditScore</th>\n",
    "      <th>Age</th>\n",
    "      <th>Tenure</th>\n",
    "      <th>Balance</th>\n",
    "      <th>NumOfProducts</th>\n",
    "      <th>HasCrCard</th>\n",
    "      <th>IsActiveMember</th>\n",
    "      <th>EstimatedSalary</th>\n",
    "      <th>Exited</th>\n",
    "      <th>Geography_France</th>\n",
    "      <th>Geography_Germany</th>\n",
    "      <th>Geography_Spain</th>\n",
    "      <th>Gender_Female</th>\n",
    "      <th>Gender_Male</th>\n",
    "    </tr>\n",
    "  </thead>\n",
```

```
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>619</td>\n",
"      <td>42</td>\n",
"      <td>2</td>\n",
"      <td>0.00</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>101348.88</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>608</td>\n",
"      <td>41</td>\n",
"      <td>1</td>\n",
"      <td>83807.86</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>112542.58</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>502</td>\n",
"      <td>42</td>\n",
"      <td>8</td>\n",
"      <td>159660.80</td>\n",
"      <td>3</td>\n",
```

```
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>113931.57</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>699</td>\n",
"      <td>39</td>\n",
"      <td>1</td>\n",
"      <td>0.00</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>93826.63</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>850</td>\n",
"      <td>43</td>\n",
"      <td>2</td>\n",
"      <td>125510.82</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>79084.10</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
```

```
    "      <td>0</td>\n",
    "    </tr>\n",
    "  </tbody>\n",
    "</table>\n",
    "</div>"
   ],
   "text/plain": [
    "   CreditScore  Age  Tenure    Balance  NumOfProducts  HasCrCard  \\\n",
    "0          619   42       2       0.00              1          1   \n",
    "1          608   41       1   83807.86              1          0   \n",
    "2          502   42       8  159660.80              3          1   \n",
    "3          699   39       1       0.00              2          0   \n",
    "4          850   43       2  125510.82              1          1   \n",
    "\n",
    "   IsActiveMember  EstimatedSalary  Exited  Geography_France  \\\n",
    "0               1        101348.88       1                 1   \n",
    "1               1        112542.58       0                 0   \n",
    "2               0        113931.57       1                 1   \n",
    "3               0         93826.63       0                 1   \n",
    "4               1         79084.10       0                 0   \n",
    "\n",
    "   Geography_Germany  Geography_Spain  Gender_Female  Gender_Male  \n",
    "0                  0                0              1            0  \n",
    "1                  0                1              1            0  \n",
    "2                  0                0              1            0  \n",
    "3                  0                0              1            0  \n",
    "4                  0                1              1            0  "
   ]
  },
  "execution_count": 13,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "df.head()"
]
},
{
"cell_type": "code",
"execution_count": 14,
"id": "36d0392b",
```

"metadata": {
"execution": {
"iopub.execute_input": "2021-12-21T16:51:27.353934Z",
"iopub.status.busy": "2021-12-21T16:51:27.348364Z",
"iopub.status.idle": "2021-12-21T16:51:27.815899Z",
"shell.execute_reply": "2021-12-21T16:51:27.815358Z",
"shell.execute_reply.started": "2021-12-21T16:22:22.129135Z"
},
"papermill": {
"duration": 0.506347,
"end_time": "2021-12-21T16:51:27.816051",
"exception": false,
"start_time": "2021-12-21T16:51:27.309704",
"status": "completed"
},
"tags": []
},
"outputs": [
{
"data": {
"text/plain": [
"<AxesSubplot:xlabel='CreditScore', ylabel='Density'>"
]
},
"execution_count": 14,
"metadata": {},
"output_type": "execute_result"
},
{
"data": {
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAAYgAAAEGCAYAAAB/+QKOAAAAOXRFWHRTb2Z0d2FyZQ
BNYXRwbG90bGliIHZlcnNpb24zLjUuMCwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy8/fFQqAAAACX
BIWXMAAAsTAAALEwEAmpwYAAAWi0lEQVR4nO3df7RdZX3n8ffHBCIKKoSoFLDBgrapWsWAO
rV2VUcEOyW14hKsFVs6OI50deq42qizKKWuruLMSGcqVmilw2CnBG11MtN0okjVjnUgQQQUMGIiAE
mA0/BgQFULgO3/sHTkcnntzudydcy95v9Y63H2e/ex9vjv7cD93/zjPSVUhSdK4J026AEnS/GRASJKaDAh
JUpMBIUlqMiAkSU2LJ13AXDnwwugRJm
pIBIUlqMiAkSU0GhCSpyYCQJDUZEJKkJgNCktRkQEiSmgwISVKTASFJajIgJElNgwZEkmOTbE6yJcn
qxvxXJflKkh1JThibd3KS6/vHPw+8DLgaO
D3k+w/VK2SpEcb8gjiaGBLVd1QVduBi4ABzqugp4aGzZ1wGfrao7q+ou4LPAsQPWKkkaM2RA
HAzcPPJ8a982Z8smOS3JxiQbt23btsvFSpJ2bNLLlSNITypABcQtw6MjzQ/q2

oZeVJM2BIQNiA3BEksOS7A2cCKyd4bLrgWOS7N9fnD6mb5Mk7SaDBURV7QBOo/vFfi1wcVVtSnJm
kuMBkhyVZCvwJuDcJJv6Ze8E/pAuZDYAZ/ZtkqTdZPGQK6+qdcC6sbbTR6Y30J0+ai17PnD+kPVJkqa2
oC9SS5KGY0BIkpoMCElSkwEhSWoyICRJTQaEJKnJgJAkNRkQkqQmA0KS1GRASJKaDAhJUpMBIU
lqMiAkSU0GhCSpyYCQJDUZEJKkJgNCktRkQEiSmgwISVKTASFJajIgJElNBoQkqcmAkCQ1GRCSpC
YDQpLUZEBIkpoMCElSkwEhSWoyICRJTQaEJKlp0IBIcmySzUm2JFndmL8feyW5lvfteyW5IMnV
Sa5N8t4h65QkPdpgAZFkCXABcCRwMHBikoPHmj0FOA14NWAZdDLXcAHk8uAJVX1Qzjbdk6F8Kb
aGBLVd1QVduBi4BVY31WARf0vdMxpRMkQAFPTbIY2AfYDBsT
MwM0jz7f2bc0+Vb
UDuBtYShcW3wduA74N/IequnP8BZKcmmRjko3btm2m2b+y2QpD3YfL1IfTTwIPBjwGHAv03y3PFOVXV
eVa2sqpXLli3i3b3TVK0hPakAFxC3DoyPND+rZmn/500tOBO4C3AP+rqh6CXwJWDlirJGnMkAGxGxA
TgiyWFJ9gZOBNaO9VkLnNxPnwBcFVbUp
yZlJju+7fQxYmmQL8G5g562w5wD7tlEFzR/WVVXDVWrJOnRFg+58qwax6wbazt9ZPp+urtp
SdpI5utFaknShBkQmqQmA0KS1GRASJKaDAhJUpMBIUlqMiAkSU0GhCSpyYCQJDUZEJKkJgNCktRk
QEiSmgwISVKTASFJajIgJElNBoQkqcmAkCQ1GRCSpCYDQpLUZEBIkpoMCElSkwEhSWoyICRJTT
MKiCR/m+QXkxgokrSHmOkv/I8H4YE2SpgRpohgRgFRVZVZdU1a8CRwA7/I/I/YE2SpHlgRgFRVZdU1a8CRwI3AZck+ackv55kryEr
CRNxoxPGSVZCrwd+q8B/oguMZmzSRpohbPpFOSTwHPByPBy4EfqmqbutnrUmycjajEM6OAA
P68qtaNNiRZUlX3V9XXKAeqSJE3YTE8xfaDR9uW5qqSZk2CSbk2xJsroxf0mSNf38pcIHc9wnE9uk3
JnjzDWiVJc2DaI4gkzk3wYOBvZZ8J8hIgYOBvZJ8hIg/aynAU/ZxbKLgHOA1wJbgQ1J1lbV5TSPdTgHuqqrDk5wInAW8W8Ocl
i4OPAr1XVlf31jwce++ZJkmZrV6eYDQJLgJWAaMBsQo4o5/+JP
DhJAGOAa6qisBquqOXW2IJGluTRsBcEGSN1bvzfzyzyUVHnDS4Hn
AZVkPbAMuKiqPjj+GklOBU4FOOjA3e9yYmbTUWHk8tpmQ5oJPy8uaPkTMBAGmCQAjMZVTvhEgKaxR6Dn6NuBXeyF
W5oqo+N/b65wHnAaxcubIGSqkWS9ki7uN7nB7/AbjG7A0xVtAPaYDAdmCQmi+QNwlb8JIMmCDnbwfJ8C8GNmGAmCiLm...

Ot2CnBXVR0OnA2c1bffDvxSVb0QOBm4cKg6JUltQx5BHA1sqaobqmo7cBGwaqzPKuCCfvqTwGuSpK
q+WlW39u2bgH2SLBmwVknSmCED4mDg5pHnW/u2Zp+q2gHcDSwd6/NG4CtVdf/4CyQ5NcnGJBu3bd
s2Z4VLkub5ReokP0132ukdrflVdV5VrayqlcuWLdu9xUnSE9yQAXELcOjI80P6tmafJIuBpwN39M8PAT4
FvK2qvjlgnZKkhiEDYgNwRJLDkuwNnAisHeuzlu4iNMAJwKVVVVUmeAfwdsLqqvjRgjZKkKQwWEP0
1hdOA9cC1wMVVtSnJmUmO77t9DFiaZAvwbmDnrbCnAYcpyf5Wv945lC1SpIebfGQK6+qdcC6sbbT
R6bvA97UWO4DwAeGrE2SNL15fZFakjQ5BoQkqcmAkCQ1GRDQpLUZEBIkpoMCElSkwEhS
WoyICRJTQaEJKnJgJAkNRkQkqQmA0KS1GRASJKaDAhJUpMBIUlqMiAmJpMuQJKmZUBIkpoMCE
lSkwEhSWoyICYlXoOQNL8ZEJKkJgNCktRkQEiSmgwISVKTATFhm269e9IlSFKTATFhF11+86RLkK
QmA2JivM1V0vxmQEiSmgyISekPIJ7kgYSkeWrQgEhybJLNSbYkOX2K/u9NsrZJ8+ZUmWj8x7b9++Ocn
rhqxzMrpk2P5gTbgOSWobLCCSLALOAY4DVgAnJVkx1u0U4O6qOhw4GzirX3YFcCLwLLw3YFcCLw08CxwEf69T3
hbL3rB5MuQZKaFg+47qOBLVV1A0CSi4BVwDUjfVYBZ/TDlwNn9T9rz02fYPEShP9z
xQz5z02fY/uD97LvxOva643sUM/jLvh7uUyP/fXh2PartR7ZeBfwKL/7Geaw/e02jQx4xXFNGL2r3q3wswz
llji6Kz+jfZYLqMZc3v7en89j23fwZ5muKQqYr8DHXPruN/d5Rz2PH0qfNatn56AVLX8BPLf2pOV/vkAF
xMDB6D+dW4GVT9amqHUnuBpb27f9nbNmDx18gyanAqf3Te5Nsnk2hT3rKk5629zP3PmLRQ7D8O7N
Zw+zcutdJ/NgD9+y+FxzAnQ/u4IBFQ76Nhuc2zA+7cxvu3A/u2nfu1/vg9x5k0X67/2THQ/c/9L3tt22/bpaL//
hUMxb0O6qqzgPOm3Qds5Vk451VKyddx+ORZOOtDzzgNkyY2zA/JNm4/fbtC3obRg15kfoW4NCR54f0
bc0+SRYDTwfumOGykqQBDRkQG4Ajkhg0x8XVGZUUUhyU5Krk3wtyca+7YAkn01yff9z/74A1
+E2zA9uw/zwRNiGH0k99ts+JEl7AD9JLUlqMiAkSU0GxATsagiSSUhyU5Krk3wtyca+7YAkn01yff9z/74
9Sf5zX/9VSY4cWc/Jff/rk5w80v7Sfv1b+mUf98e5kpyfZHuSa0baVpyf5LtJvj7SNnjNU73GHG7AGUlu6ffF15K8amSZd/e/FlC8ebLjUexuoaeAKDQJvPzumSCF88fe0bmuqJuo38yWfmSCs6Bu2yWsM/+hXnsRe51VCdQe/JQ+XWsm6Uv5cukQHyDxUvUmxlLMWkTQcd9P3B2
aqd5T/U0il/Xta/obRqYd5mYW23Bokn9Ick2STUl+u29fMPtimm1YUPtizlWVj934oLtg/03gucDewJXAinl
Q103AgWNtHwRW99OrgbP66dcDf0/3MdaXA5f17Qc8Azzlw1pwQbfHoWwHMdl32ufhbLTNYD+/+/NpV8TcNtjvW0TrCsf3ozRg36oULq7Hd0OA2pKPgE3Mr1Vp3 yaNEDBlbfHeTfqrUpkQVdAu7Sm786ap3
qNOdyGM4D3NPqu6N8vS4DD+vfRouneU8DFwIn99EeBd/bT/xr4aD99IrDmcWzDQcCR/fR+wHV9rQtm
X0yzDQtqX8z1Y+IF7GkP4BXA+pHn7wXeOw/quolHHB8Rm4KB++iBgcz99LnDSeD/gJODckfZz+7aDgG
+MtD+i3+OsezmP/OU6eM1TvcYcbsNUv5Qe8V6hu0PwFVO9p/pfprcDi8ffezuX7acX9/0yR/vkvwOvXYj
7orENC3pfPN6Hp5h2v9YQJI8aRmQCCvhMkivSDWEC8Kyquq2f/ibxrf/r/As/rpqbZhuvatjfYh7I6ap3qNuXRaf/r
l/JHTJo91G5YC/6+qdjS24RHD3AA7h7l5XPrTIy8BLmOB7ouxbYAFui/mggGhnV5ZVUfSjb77riSvGp1Z
3Z83C+qe6N1R80Cv8WfATwAvBm4D/uMcr38QSfYF/gb4YF/gb4N1X1iEHGFsq+aGzDgtwXc8WA2P3m5TAi
VXVL//O7wKfoRs/9TpKDAPqf3+27UN07Uf0mgfwu6oearXmBNV9Z2qerCqHgL+L+nG5b7gCek
W4Ym/FtmGqYm1lJshfdL9a/qqq/7ZsX1L5obcNC3BdzyYDY/WYyBMluleSpSfbdPi0cA3Rv7ydRw6FcjLde
Vn69rf1d6O8HLi7P8xfDxyTZP/+UPwYyeVc2x83VbQ9A031P9X9T/QDeMzXitUw1zM5t6QzcqwrVV9aGRWQtmX0y1DQttX8y5SV8E2RMfdHdxXEd3t8P750E9z
6W72+JKYNPOmujOg34OuB64BDigbw/dl0F9E7gaWDmyrt8tP+8GQ8zBxfhgL+mO+x/
gO6c7im7o+apXmMOt+HCvsar6H55Hs19PZsZuRNsqvdUv28v77ftE8CSvv3J/fMt/fx/fznPo5teCXdqZ2r
gK/1j9cvpH0xzTYsqH0x1w+H2pAkNXmKSkZLUZEBIkpoMCElSkwEhSWoyICRJTQaE9khxfEjeNp1J7Ic
XWZfkebNc139JVJVvTT7xvr9/5+pNCr+pFBX/b4t0QazmBfOSrNV/2Hoj4FXFBVJ/ZtP0M3js91fP
F9fC4OTNWVb858vR9wB/163sF8C/oRgy9P8mBdKN9Pp7tmFWKN0S8AD1TVR3VNWVw
KIk/5hkLXBNkkVJ/n2SDf/e+AH32fwYfTjfl/CfDMnetJ8vkkK5P8MbBPf6TwV3V3f3Vd3aL
3NUkn9KcmWSy5Psl+TAJH0KbmXTyqP5sl+TJSf4y4JcgcgL/R9355kbZJLgc/1n4I/v1/uq0lW7Z5/Qu0JPILQnugFwBVTzDsS
eEFV3ZhuVNu7q+oJOeAJEuALyX5n8+n+06AZwHXXAOePrqSqVic5xrapeDD8aBO70JNfRfeJ3TVV9oR
+OYQ3w5qrakORpwPrPR+pwA+B3+5WUy9M8pN0I+3uPAV2JPCCiqIqrozyRR/RDc3wG0meAeV5JKq+v5c5/ENPz2Z
ASI90eVXd2E8fA7xo5/UFukHUjqD7kp+/rqoHv7anVVX3JNkp8HN0RzBr0n3Bt3bBXAbVW1oe93D0

CSVwJ/2rd9I8m3gJ0B8dmqunOkxuOTvKd//mTgOcC1s9t86WEGhPZEm3h40LRxo395B/itqlo/2iEjXzv5
WPSB8nng80muphugbaojmemM1/jGqto8m5qk6XgNQnuiS4ElefiLkUjyIrq/7ketB96Zbhhokjwv3Wi3XwT
e3F+jOIjuiKDlgZFln5/kiJF5Lwa+Rf+NaEmO6vvtl27I538EfnXn69IdFbRCYD3wW/2Fd5K8ZIb/BtIueQSh
PU5VVZI3AH+S5PeA++i+cvXTY13/gu7rQL/S/wLeBvwy3R1Qr6a79vBt4MtTvNR5wFVJvgJ8CPjT/jrB
DrqRO0+tqu1J3tzP24fu+sM/Bz4C/Fl/pLEDeHt/99P4a/wh8Cf96zwJuJHubinpcXM0V0lSk6eYJElNBoQkq
cmAkCQ1GRCSpCYDQpLUZEBIkpoMCElS0/8HJ0txqYVxJvwAAAAASUVORK5CYII=\n",
      "text/plain": [
       "<Figure size 432x288 with 1 Axes>"
      ]
     },
     "metadata": {
      "needs_background": "light"
     },
     "output_type": "display_data"
    }
   ],
   "source": [
    "sns.kdeplot(df['CreditScore'], shade=True)\n",
    "sns.kdeplot(df['Age'], shade=True)\n",
    "sns.kdeplot(df['Balance'], shade=True)\n",
    "sns.kdeplot(df['EstimatedSalary'], shade=True)\n",
    "sns.kdeplot(df['Tenure'], shade=True)\n"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 15,
   "id": "e618512d",
   "metadata": {
    "execution": {
     "iopub.execute_input": "2021-12-21T16:51:27.881579Z",
     "iopub.status.busy": "2021-12-21T16:51:27.880941Z",
     "iopub.status.idle": "2021-12-21T16:51:27.890915Z",
     "shell.execute_reply": "2021-12-21T16:51:27.891473Z",
     "shell.execute_reply.started": "2021-12-21T16:22:22.601501Z"
    },
    "papermill": {
     "duration": 0.044285,
     "end_time": "2021-12-21T16:51:27.891654",
     "exception": false,
     "start_time": "2021-12-21T16:51:27.847369",
     "status": "completed"

```
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "stand= StandardScaler()\n",
  "for column in ['CreditScore','Age','Balance','EstimatedSalary','Tenure']:\n",
  "    df[column] = stand.fit_transform(df[column].values.reshape(-1,1))"
 ]
},
{
 "cell_type": "code",
 "execution_count": 16,
 "id": "107b37b8",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:27.958659Z",
   "iopub.status.busy": "2021-12-21T16:51:27.958024Z",
   "iopub.status.idle": "2021-12-21T16:51:28.378377Z",
   "shell.execute_reply": "2021-12-21T16:51:28.377793Z",
   "shell.execute_reply.started": "2021-12-21T16:22:22.615074Z"
  },
  "papermill": {
   "duration": 0.454954,
   "end_time": "2021-12-21T16:51:28.378530",
   "exception": false,
   "start_time": "2021-12-21T16:51:27.923576",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "<AxesSubplot:xlabel='CreditScore', ylabel='Density'>"
    ]
   },
   "execution_count": 16,
   "metadata": {},
   "output_type": "execute_result"
  },
```

{
"data": {
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAAYIAAAEGCAYAAABo25JHAAAAOXRFWHRTb2Z0d2FyZQB
NYXRwbG90bGliIHZlcnNpb24zLjUuMCwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy8/fFQqAAAACXB
IWXMAAAsTAAALEwEAmpwYAABHq0lEQVR4nO3deXydZ3ng/d/1nH3RvlvyIlm2Y8d2NmcnJBACo
VCWwjsspS1lCu106Ke8Zaaly0sppZ22hLbTQjtQhoHSThMgQFNIcBKyOSROvO+bbMva9+3s6aS2Rbt
o6ko+XoyNL1/XxkS+e5n+e5dCSd69y7GGNQSim1elmFDkAppVRhaSJQSqlVThOBUkqtcpoIlFJqldNEoJ
RSq5y90AHMVWVlpdmwYUOhw1BKqevKgQMHOBowxVdmOXOXXeJYMOGDezfzv7/QYSil1HVFRC5Nd
0ybhpRSapXTKCUUqucJgKllFrlNBEopdQqp4lAKaVWOU0ESim1ymkiUEqpgVJKrXXKaCFROek
I9hQ5BKZUnmgjUrDoCHbz1e2+lK9hV6FCUUnmgiUDN6oX2FwB46uuTBY1DKZUfmgjUrJ5te5YH3
I7tbdhQ5FKZUHmgjUrM6PnOe+hvvoCHYUOhSlVB5oIlAziqfiBONB6v31xIxwolwoUNSSi0yTQRqRn
3hPsrd5VhiUeGpCeso4eUWmk0EagZ9YYZ7KXeA1uLtdhpEqtQJoI1Ix72Ix72F2uk8mg3F2uk8qUWoE0EagZ9YYR6LjcnlNbhK
GIoOFTYgpdSi00SgZjQSG6IUQSA6ICUUotOE4Ga0VhsDJ/DB2QSwWhstMARKaUWmyY
CNaOx+JVE4Hf4GYuPFTgipdRi00SgZhSIB66qEdQEqTiAYwxBY5KKbWYNBGoacVSMVSMVSMZLpJC6bCwC7Zc
dpcxJO6jITSq0kmgjUtMZiY/idfhyWAwCfw6dNQ0qtQJoI1LQmmoUmeB1eTQRKrUCaCNS0xmJjeB3e7H2tESi1MmkiUNOaHDRt4nc4afjtWL
L/Df9VjOnJIqUv7+PR6wqi0AigMd+zN+4GSKSKcQFNUdVxoLkZuiPbJ25kGGIaNAmjtNGKt8IsgmC0kdr+Snz6g7KkyEg0aGHOr5vZW21fyOXHZUDAWGJQAFHNWSQ0ph3OPkKmiCKrTgS7Fi2nmHJgvY2lBIiQlbf4dVnaGoeRyTATWjd3GG36N8vDOLeBkUwqHZG0KBHYlmSSIe+c/MTCaZQDfEe8P1NDNJcwM3Ld16+nRrNBOhXMAkU+3SUHDqvQRU4i0/REX1ECNaTw/Xn/JiuNMoH0SRZUXPmAMu0XSCM1RvPhYuUfJNVDYgbQVK9xIb0BCPNgNj96KFlgdJ2M4lmgEr1bgdlFDsVqb4L6RB94LFuHNVvHxTIqdTDFkXmXOHg7prvPy/FR4d8kuAoafZcqBbwpcOBaZ/f59W1oNP8g2PUxqZ11IbRoSmwLNuTHNZT6uB86LhIYTmfRqSyGNXyJUqjS8GdP3NOu8XjB5JTMU79wp6n9pf1cJfAG1bxr/vRk9HgI+6CbmJT6L56GYUC0D+h1JP4UGDqKcRKHq8HiSW8sUVlRO7TkWUZOkpMtJ8aVsK48zqeULSiDzoRrtoSpoFJwATv6VGn1lmR1W5+ulbDM+OSjSdhg1dBbjGZDCIwHeHxcqvJIRRxUnpjYTXYhF4j9OVbYDMuaE7h1ijMmbZJrdL1MpC4RCdEp48kbUyQeopnpjGz50nvV4+6TxsW4O6iGt10KgVS+yO2/uBuvoqbMO98NWo3BdvAi1gNxKv+VDEm1q0vDrxkCcKQxYkAjCLAEVz+JV9KGhcYeYfmvIlifpeAOnkdfkB7d4+o9IXGk7pHB23x3PJzpeKJqIzLHQ8K+u0rOfzzXcfJp5xS5eiuA/TUjXIi5h9BZYnOeMbMibp9uTwe5cIUkHwO/TzuUXkbFzZKCfy9v3eSM8ztnyY3I6g1R8AftJMZpkPk5IX3JWw+emlFe6z9PYeDJsgZnIENh6MeUzEmUZIy4o86aJmvgERcT5uoy85bKkW2hY5mcfSxGBzS3K1JFcFj43DQoTdLKhtqgbrN/OH49k4/M2UDJhGxvK2X66hwRaNoNKbfEwTZ9LaH/g2jNYFuNHh52yAYApQzDCZidbF/A1hcvf6s4CiIZ07NHZMkh/SnrBXmt6Ii6LYTfqjiWDFBJTZe2V1yHU05DbfPS+Yuv4rCeFChYLO2CBxx4Y5s6i5/rIAD0j2qxmXbO9T4pIH3VakmDsXqqXFF+wrTxzwFQ7cowrFIcDe6tX8q8wjYuJZfLKHTDFFsvBgBDUNk+lz4WkO+/ZPbjn2frpXn4Ekfi18TA9ZHQ/JbyTxF3+dGyHWjDKwIWyEgY4Qjdi8CCYGNTAS5vjNzJvjn1uJvmS3WIsIv5RfnYSNNIa7/RSkmeNeGeXXnsFYpk5BmnAbcL1USvPb7Mu/FHYR+MUotT8dDN5+nP8LbfvvbWnafzFXiEWjOdYIPOHX3Kgqwo1w+SsZP8kWyQIBcH0mEmmOvgeDJq+A8Vgcg2WeQe7/+DEsTDJmbKOJG+EiD6YIWFoQ9B1cfRpPOTvw4DgCL1xxyoBahytYnqZTSr0pu+ipxBh4gtPvkJ80sNQjrNcfAAcETnyRsjYU8JBG+GVgLSW3MqFcKPQUh6aWOWrTHPFdCmacVbsBbQM3bIxrBTX/ugTx6ooo3mwL/ALhu+3G2bsAJw4A5Hj/wNekJNGsJJEGZnLzNWkRKAZvrPh0V6Ncx+9XN/DSfw50lHQypyVa3HFSTRTXx9PhjpEY/ZiuufJSqmkQE1eOQUWdbyqMJOJOsZDJAjEq/OhSggmGNdcLVxKPWG+P1aw5j16kGqvIMUjGe/Ge3RTTrDdaz0e2gRe2ia0KbR+l5jRMkPOJnESyk/BBW4qlj2HBqs7oiRmlyH2nNNFOCGeenMm0zOpcntfkSp9Gr3BmjOmHwdS8DRn2yW8FjNw3oqc+aggXBX8B8rSXyVc4wr+vUiqb8kMVKO4LHb8mgOUb3YixphlDXT7LzyxcrXanA6y/Np2A1X6sG2FoCGpk7t8Z/m/ivUFt5tm6HGoC16N4YfwY8GHqw/8a2MKjEmgqE7Ab2U3HYqCBvWpV0h4HWKHpvUwJVt9AUJbr5hNyeHRNs0XwJkbVKpl8Fw2LafxyBOjhoqCY9ZZOUGt2lNQZJ4rg3AzR5qIyzl5n72J/kt3+tzoNZC2qQB1IxUCUt6qffeDyYhM45ZdFRmdqiOk5GbjC0t5a8tXKf1XyKv9bm81FIvddV/4gWhmr0XTiAN1GMMeEe8Scv36GQBQnNdxjl53L/MTHLvu0l3W/dWVq1bUnkkmGmbhUoaq1hOu0Q2+4YeVZ4PVQbvtexaiiwSYbgsqSMB6BzzHNo6RQ37ryUsEJzL7fAW3RQLmMTvmRyXN48oHdjuQvpB2AdRmOcaeOaMd3+2YjpS5ikhsBHYBfmuhMJMTV4pSAFbgAHc7umw7lg9qt4UTnEdeE5GoGiR1nN6X2KKxqVutl+lqi5L5Ncx8+LK84Mdrb6xKJD4/IGwRP/ZmS0q+VJ1LrLZd/0JUL9VtRsF8ofadYyXr4CbJUVFEAI+DGcfMJuEjP2aTWwBTmyo8ZP+4GYKZbvFJQLjcj0vfB6Qhcq3BJG+aS9TVM3ghWAq5qfNmC/pBrJ6yYW8sdJXJV2lB4aAPHhzl10kD7zVRrTwBhvyD8umgmm4sCQgtZLSJtUCHAPAP7oHqQKNNo5vDr5rxdFFRNGBhCUN9sd7EHc7JwXzP5vQJPeyIrvLFrTvCHX4vH5iF73cA+JGCVmH/zuf4PDdqRjlz8ZagvGdgufVxUINJ9Y3O6g3sb6XSsDwZSnxHmJ9X3DftVGaF2pvCn5n3lx8fhsVHqJ1nqjf3sBzlX/BFYuHg44RD+D0+p6vYlKPtE+mOhp5CoZJMy/kR4QrpJxe37mXUSzBbbjG5SkSlfw2EjlNqlq/hTWV8eKvPRwFJ0Kw8oikgU2o8Rks8pS5sfGjFN8+2nI4UELvRTz3w+fkcydDVaCWRK0vuDNLsd/s1gH4sWr30RwKqJ9L39PJ2+Iyl3eXXN6nz2I7IqMSuiatevDcSKO31ArHtgpCsllYtCg+c14lG/3vIJRENcuuqHSWtyDD36iMAw/rRmwufwjRqyIfyB4OAvBkQ0EQs0Y+HzP+tB9PSm9ENeV6uY/c72ZV4lTh1yH0t5mPCOBC9shswD0Z5yVcZuQ2czjCvYpenfFqXLcdL8JWl7mpDFcRu+o3IJsGs79Pa8/b48GJN7hFqi55uxjqrkcoVRyrnp3qdAyL/ySSX5dSmNnagdsGxg2tiEl49TuUpebBY2hHoYIp4hjPVmj/dzDGcg8CVB1I7pSQTvqMe2YoIf2Xp6VlbcqkL1bctIjH1fLI5cCGPPdtY3I/pmqbF0+gwmZNEjTf3H36ZTbMbbxKjw4u9mSP3gExXTBtCa/wLHPepMzKXvbhEzPYNz8cBxTiPt89HKZr9NfSWjDh6xQU3xZeoUGDAOn+9/yv4iZRWT1RqjfCiWjZcPdqLPiaiK5Jk/2cuxeb+W7hSqLWfs5Xk5oIJMSSrmF19e54vLlSPsrRg0a4T2DVfoO85J9txFzHBFSmzj26pn14l4I3l+o1wB8mj4wMxJ+wSz2i5wXtyfFHz0Vk0ByXvnnm/Gt2Rqq/qnUE4oZh7bQN3Xsau0tsd6FbLLaXlsKCrqHYzFBE5Qmm9Z9CBq8N2R6wJXNxU3jBKtiWvzNcvlPIBi6WV3JzKo4+ccPI1CY5hpnoZw75ekMkqtkKNRgaCrd/2nk4Tbnf9WBhfHV4pH/PuWnCD9vkzuU8CWyEzArqf9+86PJ6w6qvy4hlOgfD/cY1c+yHQXGzVs3JoO1AiwlQo1Gl9r3TTuknXSbNeCj60q4GqxH/+4n7Dh2LYm6R3+PPp8gP2jE0rMqSQ0jV4FeyrOpj3UUsqsaOfzO6XGaZiOy3iA+zVHXm23ruUb1aOu21mgUrCK4ArDuX1d3VKgkvFfu/8s8+7aGC5t3tXrzh3xBdVdL2dhGM2v0Kaq/5XPMZNgWbCMZfx4uK1mn/6aXN40H4mnU8aNzZBnTpibOcTTTKN2fRVuDctCCRBvlXiOvhpl8XJ5HkjjvOg4IwDqRxfxtwlXg3x2sqLrMHGsmbiQ2KJkGrGEUBmgZVvwlUEzlGyzvf8btU0oaGKYKtidIFRBAhX8L/mNbcy6k/W+UTPA37e/0a12qj+NMT5hvsLmjJqN48d4LvFf8Q4s8W9Aqjnc5Tp0hWm+v1w76QV/CnXaH1zbhp/7QHh6xnlrxU2KrCGeXGcUqAiGaEl77uNy5ku0fLYgKdjAPzwWp7VO8DFxVnAa0i7LbF9s1+hQfqfRmm2XCEm0MeSn48z+5xBfjV1L9Db1WiNVD4ZeKN57x3W+VXxQvAfqsc/b3prfY1sn/H2nS+Q/LRjxMFVrhQ6ApfMOpbxsc5LIOfoxH3zkjgL1otu4Xu4/j9rZ5l7nEOtzPQ5OPXjx/nlz4zJwmCAcImjo9CKq6hnw8AkkeOeSnjBKsoN/MWhYt8bs5oEvzB8RHSrSfFZQmhQ2ERLnV4ezr67YNdGf+w/V0R4x+QQlWq4UADkb7oyiATr7bC3uyXlzPIBH+OfUdeSpZSG8EgWjY86f6yTONhXlOZxGtzORs6uKNYG7u72zkaT+9N9dsdvaI0Mif1DRnZ/hPqeOpp+8hKmx4aqd5DyljYHTsMmkT2Vdwp4FWs7yDq1UWTgXrYYQKm8eB/DwT++c0CWn/kqqf36eO3ZtJ9tgoovtR9VIfpH3bfa8Ie3xWc5BfN3nTB71fq3HFwfyA51knO6kTzCtI1VptW1ucNhCt2DaXwY8uX49Lw2nR2OiP5Oa3YfP3knxK7OmaWmFBHgO/8FGh3o1yM8qw5pmQbLqE/QoY3BfGmMDvGYuJbTQkZePG48FmpVi/w9mTsWJmeB20Or5xyvDpe/CT5ZvUbAzdrSd6irmAi/ZxvCQU6zH2CfiDPK6A0F5gdUWFupwHG5ks6UNDdkN4AATE+yTbNOg7v2thbzdiULp8MkTtczjdamCIGj7OCHgD/cMdl38lqPKz0u7WBqVBvdtNiVqUg0t/YVZUJGoPqNqyOn7KMEKX7tnWAVWRfzFp/3WF07UJ6xqVIUuWfR6jhuhW1NDv1mX6fg7PR7/MAOqrekjYP05AFQCBbSQQjxTLBCGAbD0r/NIWMl6tLLIH0m8yIq6iVzxr2RYGYaz8yQE/h5/1AAj8eq0OeQzhgZ4zPy+UiavUTWNxR1pZ6Yfw6ZuvTaWw+5ItUVGZDxoXAnfEk+R2/jSm1gRu8TzoKUkOJvakiSzO5vUgu1+3FS44/3fKXeIGP6ONKddyEXNcZ30sUd6fnyUHJSZtjcydgkGFjrX3G2NSRKKgCvbHcmLbwo4sW2AH/ai+8s+EYjkPj3Z9DmzeIgm+S1VPYDwdFL+Yw6u8Hwfgq7tHvbKT9uXdZlRqJp3a+HTm+xfe67+Oxu+BeYr4NHy+JXq0Z8HoGFEs/dc4fvBBBoNzNcRWuewqVWxujgyPxkY8E1Pt9NEgaxX5Thq4HVZIYq6SRTzvwmfNKKfx7Q6rAD8MHIeJPrjmu3+dvhZj0fc6R6TOYnZpi/8SS7GZI/TWBO7OP8fQHhSFeN/lVxr0XZ88vVduW+8UNy40gVO/4WiHHpoUx+Onu7rgfxrm2kdPl1/3nbW2lOyXiujJAqD4Jl2Lz9UbaWPxgrjMmNm6KC5UOE7Bb9XOqbAhuWvqvjQ+RTZpW1hP0/ESuh8NXkxk0VPEm9u8PNqr5zvGpB68DlTrDSiPSiGiC0Py2BymOhlTYyGzrUFKLpGjPRjIQZAmlzY9xJzOb2cYxBiX6N7QdOnxUJ+5fbIHbeNBLdlYG7LOoZKSjrcTa+0G8P/2tyxH3x4tvnNx17uqOTmnwTpIW5y0SE/Bxs82XLjvbyfGS4VfyedvO9JVVHqlh8dbFwmrrvtgMNJkvcoI4Mc77s+o6Ed8m4yy9UjGsypxzZ7c/yzDdM2yuuYtTSlPmC9RCWuJv/Lq0gmX7eLvyPEf9PBWBNU05F4fFb29KQNt+D2o85zujl90oB2HaiCwJN4b1dVQXZOk/Y1x6uGW8FnG0DFKjrO6+JNY4R5LmDM+QIEUpyfy89kB7MHtzUvpJVpwakTe3J26p4jS8NUbm3KjsSDPCJX3Tn5fXQWsZCVHAIQrutEt7mBxMu/PmVEPaK00cCjvvxGLU3vx+05/T3zcE7zkNNW3Mzy2MvK5jRM8YNsj9uoobHxAuBP8F4U2Eln1LwBgfTowvA2ghUu3VaC7mLXwb5VJL7mivLnUvJ0G6BZuh9Rr3Zsnve2qs0flzudnZ46Ay/YiMz22ppUhqGQpDY0FMbf8hgTYoRo0iqYy3l9gwtzjRf0zYKSrMWyR1c9o0qtf2Hdi0pqoPr51M5qrwKrTzqR3NSFRQDuMNWrDdUD8RHcp9xHpG8dIvNBRoPpl6jYEL8G6/oMBS9Q+tIfWhaS8o9TCgN9k7dn6ZkcaQqWm4X8YKbf52EG0ULfaL3JflGnjKKfkFU6HNHu7GBuMJCmjx72C4IhwBljaaVBHTPwvNw7i0L4ypoMoLhJzq6Xxc6QTJkUCqibo8ylrvLGH1wjMmT9B8JVjkVmEMd2v0IHEYSm3wUeggBEG4zUR1s6DqJZ7ZL2DcljhubmTTVnd+7VB8r6gl/1aVdRRr7HHNi9+BuBu3JbPPZ3Rm66X7UhmE5NxUNA4KVYkh7nhjHdFzfE+AJb4ugVA4OaAPpf4fhrkxZjKPQ8uAWb+v5ppw3Z8RAVAN2v8EjJ8kaJ4oCtzXivI5UQc7v2ovBpNm4NEmm7J3PpY6Br9GiXa98ThhrTTw/WYx60KHgEzJ+wz6hdaf6tm9TrvxCOnLmpQSa8W2NSf54NLAo3Kh9eX4B8mjhwLiIzbqwRPsvJ6Hy2gDN0mCcLO1exKeuHkJ2o3gbgkAy64LCwhpAIu8WJ6YVCS5Cea5NknG0yD3O99FQaTNtc8Z8RcxxQCmq6BLTLIE9+zcTrNOXP6WTy5xmp9GjmtVGe6QtFnvQD7xA5tFWmakThU5yXGYtYBoLw6VUvyMLa8bJg3FLOewe76wZW/VOb6kz8bmUp31eNHsPYbc6W5bQ3l9XOPvf7e/qgDHGkxjeA1sPQZVIRcq2DvKV8eu3pqLwIbBhjDHuzQoZgJKRHHZxIk/+2aDKkPLMcSbT+IodANAl0tKgZQaOTa/rhaNaKbgXUgaidp1qFwtftLTWJ+uJCR6yUTCGiYKV2B0KuF1kTnPEmVTbxTl7t9F1vjkhw6hyAyi8sjz5tcOmFvzpvAHKmm02wH3bbb3K4vqK6dF17T+rtDZmYt7h8Sb0qZdBFa8PYkbhl5sB+sIEMf4nAUW3F73mKzlvw06TqgOiCbxAiOYbJy9YPfZaG58mGKq2K8uJLgNrkhr6rMoH6XKQhNtoFRnv8VSUsIKDtJmQwyBhcYMR8k3L2gV4GR/3x3Sfk3I4vNPJ7NoxtiulZx3t8FWr6EkflazSk6SW4r+Y+nzxEjR0Tgqw0Tqvpi/58IrkYHXgCqsaqm8bSU2qTETuVKsi5+jdD0bzxC37Y+b6R7AUNvJAf2P5F4UwQNDYkvAYxKYD2Xp9HXrTdiLtMgwQ+TCR9s04i/Ao08bNJuT+UvoXVXkUVk6o+qc+8ylxYbmlWa9PNTm0TuMK3fzlHqTkcVZGVdN0mGAVubAmUhMw8qTwZNjyKpdyOx5O4vfyj3PwRS7B62M8SOsvSPi8OyBUjI0EwDx8F9ntBVlfBeDXBiD4lQnMCOCeywvdLZ2BC8Fsqc95MuGHZLK2dKVpAe6xySdJ7EMoFzrnuNsGnTYVlvbODuxGH6xFKF8PAOaDGlSODGJbS4mH6zdN8CAXOsfkmgR7ZvtgTjd4shHA8vQc7QPmQ/KNb3zGC0hLsECVAZyS6zbD+4HmiExOQMHSeIbgbJ2U69XYFdobKn7Oj2C6TeO7JxU4MgAmFRPsNBSSTHixJmvZtIaGtQgSRqXQBXfa4C3KZqtFnCnt6D+E0hFWFh6Oc3BPMcODd56hebJBR68zJT6Plc6BSMwFepy6+ODDJgDc6jtDwJX6fW8LNHlNCPj3+2B1MNZBiCgY9hSsgIwUTzGLJrHKuF7BfXrpVrlcuNUKK44i1NWpMqCDzCrBiE5D18odS6ReD1p+KYMvI8Fh5kYnHCl6wzhOcuvUQJ0Mb+7P5sDTT3WZ06/VOMHbNPgYqOOP8PH6MFT3OGZDGZMu2Hz/Gb+FnNa08FhmO9BEbp7oTHnYz70B40hkJDFN9kcAHh2spmBWqK5CX5dGBpt8XGdlwB/8UXJiXK7Qw6rK2fcAkhEg5kSsfg9Mn4BrlOhsWtaxp0XdsJkHXS1TOC4fw3EqumyAmr3vqwOZr7pWIlW9Vx9jgS0r7W6cUt16HpW5LKr3pAXQXHEpkCSiOawMxWuwAk0CqhrhYDGWbo/wrGuthmi17vHYrDHlYIe4zaXK2oEKhptePzyhGPnnlt5TS3dDf3Esbd+qU4ko7GjnXa57Nr4F7k23YCKw9iPvdpkQgE59JkBQ1mVyfE1Y7VQZwUfpNavBM2BWNTNmpQ0BBAqIVgy5wVVzEv85d7ehFQdKnqq+FD4gOUOpsqyDT17eRR1gUoj/K6DDzyZjPrWTGZ6dYg+IxzwGWX+H2lwa55X9A8HW6zROOSrafqBIQdRMxNqUEIBZjSoklLyz61LjO+0cyh7EVyJPq3lMqcXn7E1iSDThd6CzPZ0JvbOXB/nRXAgNPgd9cJx0y8ECfgLNZK0AyvVmyB8d5YLKVmSGtgnv4avH8Ppv/WwlqaaqUQELrHvNvnzN2VZbuzbgnmz/sEBZJy5d1IwJnXCqh0LlK4mAAJVUtPSKjd2Mg8FKsYt5dTvrhV4rhRQBEM1EEo+eMvaOpPvT7zHTkC0z8hzMMF0vbwPrR0nmPcgPzvpcg9zJIACCVSjd9sFvsiqXXmnhn9pUeUm5qVWmV3nnzvlJhu2VR/x8FDoiQV0qcRTA/zzCaiQ44iD6WSoZz17gxhPv1dgx2H6oeiCp0ad+Oo4iSNgSImgFpm8mGaU9RPc3Ya7s6fSzGhsnHAebDZ6w8aBCHrIEDHRsR2IeH3eOPuy4EZiShO4OuCXiY1yUEH5Bq3wCpVxsQLjNaWkLWdkMY2Y+j3f3NVCM2P6DRRBHqLDC3OQBPuMPiXXBWq+9UJK3Lq38DDuowqHdAT3Lxq38yRjwEi1MFVtSVWzFDrp8m5dH0Z29M6Y5MdyGYIiPmJ3A3F4Vs2BSPqwrc0X4GG93xPKmHxP3jfpAQX5OqZTOvh4hZtmA1Z1lUGcGYGkNJuf4B9NbqXvXH3e4JnGv5vHt7j9S3qR9UZBXc4uWHcixeR0Ei5NV8cPtUI7ty8J9v3HfDhQUFS+SZEXk9AXzRJjIZs8dx8aLy1Iffgq+u+7FNE7SQpb3N/eLmXdxZbDR4H9dydRzGXzrKBn20bCX6y5IYR3GGZfFY2cZLoVnEVNhtpMVGDTv3PQMxYVTb+JVbMRNrFpgT/hSnEU7Py1U5EVcMFfEwmTvePbXkHtoLjfBvuIiN2G4RX6XPW8uWfKXq3oW8O9BECz+p6Q9W4X6UZIcycWYNnkeRkGEUoIJ57CIuEf4NVfDIc3bQ30oHHfF07ltmoSmlTV55IeJXf9wGSqCfeMxjyrhjx0u9fWCtl8bZc8xlMl3bU4h03hTW4v2BNnMyQ/GmXFQIfljoXx8VKSLdJcktTI7ZY8q+oEMGR29SsuIY2R78uhqxt4Tgn7DBSFmoGmG5rHLXr6lAPbXCuo6sMTe9HXKZE3jjcdxFxI6Cpph4rRg4W8XsGl0Efa+jZyUsvrd8zXPVYN4KMRzLGaLsxzHr4Tsu36ryi3rl5y14j/JzRjfzgaW8z6XKe53oi3uZ6d7BXh+CRqkE2hpE5tEDFjt/aI8UpO23LD4RGw13HF0cjQ3ZuV8TMVFTdJEQHeqmhaMkAp1dLFDc3DNROAm+GIjLHbpGV6hnGH9Dbl2dJKIobO9C7LF3uBHBVAG37oKMPs0LQIzqvk6VCvb6qVBp5uCmCVaWXDaTHCPX0r7qQSF0T5TgbXZEtmQ0R3NgnMVKRIuoq6GAbRrrTT2XpPn+JD6LCZ/lIJmmnGC6PBNl9Y5rRG0IZ7ShXwCm4u2xs3NsbzR4A7VH5SMvWl6OLgyHWHnA5qSmIV8LZlKNiUWPIChQqNtRP0xAb4GjTW3tRcAiLpAnVwaMKCdRhlE1Bm99kwvcbWyU5oHLGc0KB4RFsXzhwGLWl7bkqs3tvOJMqWrb4hUiL2tq+bvEEdKgCc+XqRwfgJQ+NXAD1PLtTGzBF2L9y8sUemXRwJmRRhVmSdCSYJLWVwz4OjnWzIbw24wrKm3mRMpMFRTQ78qbHBqf57PFdZyH+kq1gW0eGawy0jVh8tY8OAFX3VXE4w+cvQ1b1fHo7OC+Wi/J9k5+TYo4J4Vz9Sv/dnk9SJKHnw3HZPhTE8jHUzgyWrIqsGBMWdyaPRi54EhlyhIM88NkCjbrAdj3ULoUHNb/mMs8vkPrh5+tspbX4mnvvx2ELnjHIGZmIL4TKyHVDbSspg2BYMSavkBNTt5nHVtsXOAhUdYdwRWPvCGQhCRkcoHYlLGgvGc+wU7ylLkbRVwN+pbbZnHxU0+2bzfaArChGnnmafFSlPfZRpEOgJdfVYjDVaXvoeVzQO+YlDmeMCpuPcf2gLU2PldkuC/H+cOaBi6EdXhZWYbBnn5+aamPIyahFRvkSZdTYOpt0S8gnwcJeZOi6oqv70UGqdR6lgr/94QmOq7KuVHu1PbS3eApTrEHFUAmdkRXmQGJ3sHCjDcVtsB2F3GjedSmbOHP5JENudDQUhwwmZRXBBz4xbHW1vRGbhlAgXEaxZVYPYvNQmnR+eGJ4IhEqOFNY7e3w4qCTM7yYXWFGFzZ3mbxBUh9uf8yUp+EXf44SZkD0z4B7RUMyWdgQEnxQ2Bck2TkO/+lE+4vMP73swrsdzbjIq2LVGoVAFsABvhESHgQ/lnhP2nL+k96kWN1QHGJzWhCX6OJSoxALYNhThKG2eTNdmvbP9A0nBJyBIY8B0Rt7eKrFb9CEb97EBmAlScSKJ7KZ9zPejmU8w8KFYfLb8pyMBTvmNRv2w9Z8PTxpgtWSc0zhvnaF6P17s6RTBgRZUL2Hhyr7J9mGfQR1zuhnqCwQ3ukPOZRZSTdNHIeFdkA8lAMExjkwZYlrEL8kVqlzLO9hMs1J6h5DJ3dUuLE8g6cJNb5J7C0t4mJlH8pdGD1CSPlgjbZmXdnZV+JwTIyUA2VbSE5b8x4eKGmqaGMSYMRVc3yuIhTVwE9t0zGBzuXjQX9Q/P5bFrX4ud44cx/vJ6wK4nDrS3dhjG46PxWzrB2pORgWDcEKNcnCvS6egXwnbF11AONrnfm44UQJsBw+SJ/6vb4tgE7u7cVaPI1Ht6UB45OpPqXhDMa5bVpzjqDtf8jHQQxfKfPv6RMaIodD31/+SqtbH7jNgT5tmo7JKNykcyWWu+fI6HRdddRYFRRNHHZ71GUcimE9f+Zd9FQOp68c0bLSyCqwrSQQ/JkN9vWP2bYdJV8Sd3KWNoBS0Py4sfgsFmVWpudB1MeOKNcSe7ejUDHhSa5pqQ8Vhz3ALc3mzOnvZ4eKnlLk/SjJZgJ8wmC2Qy/N7ATMcLn1VqqSbQTCUAdMLAm8u7PXlrtbjDwsS87JCVm+3y6Fh6ecEMwA7iA5qt2lfu9rNzd+AAqUIyOvU9OfV3RwC8sKsS5OahjR4uh/tk/FUmMY1SLp9xlFFaVIw2zRYbvhPUGsPHbvyUH9+i8KCPJkBOH0Y/F1JjzR81WVfDz+KXtVMfWjxqeSpD4uEcr2q+dTB6xEGIE3T7IiEAuHxC/kOVrwHugdK5A7GF3pDfPEjMuL1OrvtNwqmGhZgYLHG2trd2QF9KfmMw6d6ul09CJSIqn5BXJgOZVGBUxxV09S/vuCpAowU6w0nTlbyuqdO+IvFOBDDuW8NcqN6M8d2gnnzcDwCG6YbXQEhIb3hW6ftoQsB2cO3BQxD3cZHdpJ+1jhc7xuKRhT1DGwd3Yw1t6vwfv8EAPEyOlAAhtsCf5A/bPPY9flpyOUILQ3YJk8Z9fNFNhFJlLwK2yOYpnbgbG6bJiJLeR7OsIPFKvAy1ZZAhIN5gZa8RENEmNaQ0QpDq1bStqrMdDKnpv1LT/jQhoWDOjmAAmnSoQ4s2h8KqKKl+ruKBiLCJTWrtRlhsH3n8pBo5y0qPYCxl5Z0fQhDVj+eIw2M9N2WMsY1WQe1WLcY1tsv4Fw/CMdxFdKn7+mrsyoHvz4ILGhAXmgtSXT9AVnJ12uW6mglUgjVbmJk0hTCDN2GaQsT6a8n7JdU/43g5UzKs6zBd6K1zJBtSPVNMpkZX9wN7x6KzZDr2RV1/C6lRkvsV3MlmWQxrK9tPkF0YHXJOZTf0UUmYsZhxMbljyOZwWxpRhgwrGNmHqBNU1KXCO4EsMAwHKy0xCBqJ5e9VcAqiXTYRgPKZkKzvnQAbHfGwlWG6MiyIZ1HlJYM1krIJnSNHF8drmNRoIzj5vjzk5wCRuZmNYCmzlG/ZSDzXr6qr7IqLcD2nkI7AqOkZcJ3Kyj0LHNSxeYgDnoQAeEWUwj4XEcODkfh67K5Y6PRU7IW+FZWSijvKOj/aXHwYpFFO5PUSTHngAtZxz64NZf5Yt6l+8PRCBvEGbYF+uGFlNYXdzlKNJvUoEnh3qxqT2cOtfJOXZ4UeNwh2EvhoBeTvx6aPKkPydN4rDUgo/Bf5Nov3hAmZhEyxPtM1fQsq5FRpoG89URTGDpgjEhaEM6bsJtuHIz6N2Sws5EpXiHKsb86h5k4RV6nrI2d2LLrWJtd9wUPxgKGN8LDKQ8z19dCrLmIXPK7vBvQeE3dEd8qnmHjFDUWMv5Y6RBdhvqWlCPx8TzzkR+JT21zk06Gq9SJwHnPXC4EWXHK5ubSk5Sd4ddpWQVWwgQkiW5F7bPjnu2Eft3XiVHgEzCLsuTIqN3SmptT8AaaN7k5RDDzAzjHUBKW2ElmiNVDYvg4XnslSw0aP/0AI1EDaIwY0gJsmvIfCx+k1DC1gQxkO6iJHL6YFCMsCrSvJx4A0TCsKm4ZhQ43uqmFd7jkqUhSI7tdDbljwGDSpBmrKYWySl3+7lbqwhXgKtmhbs5t1oJHvo/l6dhYs7dxZyM5GxUBtZd6P7mFdFxpcgNZjC3oQRymF7ryC3CqdXaxC5wUbxAFVkjrvfvyfaaLfd2m8JB5+6ANWCvLzFGPUdW0BbLD5s0XeCEm4Z3MgzwAtrK6SoyRMSVTw7SfhZQaAxpeYSAFnD7IQS0aYRQCrqwMF7yBpYL8fmnUvxLwPehNqMYyLKe4ClixvJjiLcXPDMq55SHhkkWVlRbq/+e2VZnzmzVwNZjJYJyrtQt9xJNr7MBGYOm5MaXE+LAQHiBqlbNjwJ1hP0/kGdX8J+KqgRxnJYkAjDmCbw3qGhGyZh3DBsHwHNbhH0sHrvzxxiJqQFiuEUK3H3hPH6M0cj7xO8szkoZtTl3PqFGTVVhZYYT5RJF8L2Ei7gzSQeQPXyNZiODXKMcWjZm6wkjGr9+I6kPQdaHXvpuyZM1wU9aU5LlhLXORKhTwq3Nm38PnGTqUIr4Uy2p9zgW5tBXTZClvFPYy2Sry6HstXf2FlPsZFf0RDljn6KaAwA8TAgh2ITjHfADrJQZdEpDdTJSU9nhKcRSW6aWoXZnfm5lPZn5FKSNtxpA5rt8v+1PnQnglMcrFqELaAtUy6xCS/bA3CjFqa8KaUGPT9QXTP8DVtfhzwhwzgVvpdBBvXqffHgLRMNHBNY5A+IL3P4uWNjfRqfsWZZfPI4fAtlEKxz7y7XyaW5DP3UeupBwETUAVbp1DN1uVgkd7uGBZWD1llR66jLiGZ5fSdRzcdAkPmTxMhsDEGo5ujP4kHkYsXUVMyKn2PU/5YmPZeJWD9MsCHNgwlQpd7P4PLyunt6+8COvnczeOlg9bS+7Qe+iuafxlE9hDXKn4pOaUQXNZfPc1Q7brwXXqgSHs+OHAXg0uKqAHbzUdr1vxBmDBuwGmWTAmHqjf1CH4OIrGtdvgdeddiNVjyuF21b9iMVRT8YyfFgvaxZ7Q7JBh3KGPlfKgWMPq0IJjSO6XZ6IgtKoP0GAuyEWZXPtC7L1BJFC1dwQp6H0Mv7Zu8TtjEybmWGVEg5aeGkm3s1ifAfKUjAG4Gz1ljgUFfCUA+mhAVEGdJ6W4tlFEWeNs+h9rXMcKiwVVZH7kMCkUdV1PoPr93sICjlt7hx1Ebqt6OoRfjhrldtRuNdB9H8fHsxnKt8Ia0yzI4p6LeJfHl4eJPC1wkxyHAHOItzG6kKeU7npzYnFJk3HxgGyTzD+HNHxvrsBAwsfvsnBjMyMp9RoYkeWBZ27hdCKPeTe1kxCGPDvW48KNlWdiNUh2I2L0cp+vjyRXTj1tbwkhQGJyWJQCLbjx1Y8Hhab/Jf2IodXXBG6+hGSZI1H6mD3WHNfTMhu2ZM4vXYo2Fww3J+x8EBsR+lc5NUPzmTt2RSWQpx4MkEfmBDjOnLvXKVVhY4JY8u0ihH2NQzkUqMWCH5lD1mgL6o1Bj1LvbJdvaNkkoJEO7Q08gdIfaQ8zFBuFCpaXxKCKkMPr8GxB11KAWPjHr2T0zlwGqvDFi7eOoHB3nmpSqnP+4lVqxHdvrV0Rz5u6APJqbBbW7aJHI+n11XVUaYDpQI9eLEiy6VgsmhLHMIUd+0RDSlEPAu3tGxD0xZO+bWQ29Vw/xzU2B7QwdlRpjMwZLdzd3e0LmCpOK1i8vOjrwRqgMTrJ+wWwcuhGvJCgxTGHd0BF6bTRoyPMp3ncL88Zok6SmD3yiBNHrrHX9dZ2wM2BY71fofKhbzhOJ7ZVbD56SRijhAHJcbkSuCl4w8VHUYtv04Sgnnc0zgg3vDiNtRyQMLLTmQZNn5Ctw6gMP5qjYxkqTt7HhaHRYcFDy1RnEcwVHPvBdm5u5rgIfVjQRBQ5tfgW4YOzf9F+GUkiRo8tzFxf6W8QlrsjfoRGkKpIgEnK4YdHn4VGRyNjAgGsu6D+D1zRg2UxuqFe71kqjv4RY2xcYp8CMEdt7kHZeIyKlzlQrZpL/FgjTTYGmxgPVvldfKqnBYpbKlGdoXTkswJjYLfHpAO3tyNk07MbaYJdHK1SADp/7AVThk5b/40ddo0kaIIyzy/3TVyWAxIwAbEHy6nA03bK8c4AiS1bkmR2JfZnN2w1UA5RsNDE4t4+N/gQs0oHpWuGTdJQC4RhaUYQa2oQFF+5b/bwJKehVAsbnkBLzvyPSQ91U8F0OhiaJOD6sQG5GyvQ1DzaG2bUDiWpZLYYYBfGs5sa8aGFiK5Vz/mMtuk15Ck2jcBoqqD8rR8YO4sTg9D6T+TIIrGOZrlbIPS9jsdqtYv3DPqcXWWRtCQd7Pqw0Rm6bZ4+IF/ijR2Rg81aG0ubYjAqCVA10sjMVy+6Pns1lVqb7yaXfZSazfw0wtNb+9wCefiwK7bZSgNeHNcUSHxa5gn63FhQQNjLq1YOHw4DETTYuzkqpPpOzojUfA+NLb6VhUIyKSmtGBPvrHD7/4Gw+eVmsyj1rnnHfTpjQ+cHOT6+KRKoKrYzmA5ykkOLapqWSx1KSmhXaF7/c8OjBMJyjFBrcS5Qat6ZUy1MSAMmAZ4dnwL6K0LhX7t1vvt/0Y4ji9Ooxd/jBBGtyMUNPY1I8nHdDGP1SiaQfUtDr81u6WXVHu52S5DbCnqqZUNe46iRq1l0Gl4ZmxH1uuyzTs3hJp2Rx5EjdnPKohZGRsJhpWCcyMi+sNkR3ifSavyU3CBjhLKIfBEgeNq6lCpXD+qzAsE5EFrfLClThd70fypwBvTlcr5Eq0+fbY0Amvwo3NmhcJXdGQvG5GD1Xc7wYbCe4R1bP4cGH8QnNKtHz7KAVNzjZtwdxyAOpIHe8Crc9NpXRRgTp7Rv6vAX7dDSW6HOrHdGLcQswiK+3xaMLnw96zaEpnhs2FyPk0VxHhOEqqWxIBz53KQzkRsjE8RvDZ67QMDl5SzCG2mFoU3uKtOJfAjEXDK5KQyq5A0DzShTzcOAKbDzoSzH1j0cR3ptQ2fDH3ALNGK2E6MzILVtkKtjQeIKFlbsFNYG0uYCsOhIxP1XfufU/Wa29K9gUwpTrH88thEVt5C1aAKhhsSPNW0fEBNYKzvuNXmnHPqgRG2YY9ru7ZrG1qoPEKudy3a3cu7Vn4x5G9aAsfnVhQqyAzEklhtlHZ8wXj8N0AnVZ3mFT5z3sq4IIj3a9XSftxwPHM8f2+8bGMt3vnZ2ASdRJomQd+sO/7v57IHTy53Zx5vNjnzKhWxV9yC1PMGiHXzY3cDbyQ1KJ6Gb7MSgU0fNPRgaBxTsrOIS4rjnS2vu6rNmytEHGSqvIDjbwiwW6YN2zxRxm9E3sddhc2c/lm6yKsvaPUPoHC4WvQYSi7hHlqQgIcXSYomNcmSLrXvLaFxR8XMMjCFoPMi+d0NjnjPUWObuuJFFjRNVc/gmW8jvSwu6QT9tZGVQXK8lZqVSP9S86jkKzC5BUemKMqqaRjNyaYSHMDsW6vBoDZD6KeVHb2gKpmgIL54YOkObFJKrk45oPWPEFEV44yvsyvqw5HOBnJnl3J3WIK3mCrCp5Anlg6lALfHtAUXm2KYT0G7OChaUE7S8tM5lSfBrmlA9OfcI8OAjrKCsy1bVPNUSlEHWiuYrTfNUsrf9PlfHgUJnlRuUNHYqhd5lO0HCmLMSsa7nsVaGWz0Rv05i4wJhYZC2vPG2q89lIYbfV5CIYtUd3t4D2eOyT6vrUjbYxRo6fE4xYK4hHtQ2CHkAKCsLT0hJ6g1nQCb0aqz+SFCm3eFNTVe3jyHp6pzP8aYUAW2FmtP5u7Ji8ofcpIumgfbPS8umI5hk4DLGOpBAkMJYuArCq3AwIqYJ3hNAvVJ1JRbUX3RBOBEGwn1K6DNQ4VlvuNdwgT1p5tz2trVcCSbDPGKeNJzO9UapwKsd7yAVj8DOCfW6utTuf0R8qVa7nMSWJY1dPe13ChXWL1CtHlNd/7V/ENoE23z4jjaNl+R8SIw6zqSkH4UfgoSSIIG6T1hAxlhIw2fFn5dgSKKS4hGS8wBtmwtYEJRyW5kbKQN4+vuYbHNnVuHdeRiJdU4K0SZmOK3Y2h7QaS6c4c15ocklQt1cmq8Zz6WxTr38SH+v7qZl14Lx1f1UTkGR36oK4ZS8qOtQUlZtIPYl8wTFvIxP6LqMXnjAX7Yh2rXmQNwc9I7nATKdqc8Tzpx3GEmrkTbm7OSVBdmVb/4vGRAt2YD0hNEB5d8KS9vyeLaJCW24b24IIsTDHx1d1A9eBZKjbg47YvfbzR+/FTCoIMkoxvBzwuVRz6ZKcTcO7bTtU5tB8oRmAXwhAAfn30K6uwhXUJZ4GfX1BRSKSMgZ/mvaEm2DRoytPTDOaTNZkI3zAJw+vT0fvJhkoCmEdPXG4cZ63LHcbvbMOq4SMuqWcrWLFwybszmw9tlYSPZhxZlNXtbN7RLZhKYi5oF7uO6eSvEgl8czEMB0tPmtV2qpbthqsBR/eU3KmI/G5Cu9xDwgmhZc/nI1JPhQKd0F+2uwoVC+XFPfjIGLh5H3KNJ5e6ZFbPg9vGeXbKUf3pLAQjnwFvL9jsJz0wtrc7M9shgVv6YmAUMjwBY3PIzMohLwt9wJYi0eiNR8s3IGJdEmVqx2xfuT0x7uAUAbvjVeLa3JxD6h0UJlKlZGgEDzTVu5zDLqHQmUXSuH9K57XXp6sJPeMvb2Z9zkEQRZbf6XtwgTfFK8U4CFoBSiGG0oE2e3D5kJxW0J1HLiAwXPaPH4WfT4k8ZmRW3LKPlASdXaOj6zqcjRg2S9cPSMl1uDmthpPd6S3X8SEIy2xAhe6uPQrC6oBiMrRzGqkAD6ibzzYWTGw0n1mZbuQFLEs4WoCcuS8IbvNvRuKd0DOuhN3KQ6YRaxKSzsP5J92EnHp3P9YqGsy5FIE5yABlwUyDJBZZFjXtpsk5yiX6EKWKsu9uf6PJ2UhPWlP0ZdAlmLGa4BuBSRAOA3kLKl2eAtDXzyaKYSxnU0V8Opl5k2NcjXIzZrZ2oLG46FpWqXAtvHL9XjaqeECAFxYQdMeuxeuCVT3DpNrbKl6pH8aqfCeqYLrUWo4GIb0fxH4OZgWU4ym00Ykw8fJcStdsUWjRl1OzawQMMSmW0GfjCeOsfxSjFlYVJ79Xl8GKAzX63uQm3Zod6oI+vwZjf2v2ZxVkCZmA3o6VmUsWhXmhErDv5MxU3QmWpZ58RwAmUJ2sXNfrOV62PC3TBGPnwWPnE1UBIJ+0JcIkSwZ9bdUsc58wRmbHK5o4MgaFUpfAu72RjG66EQMP8AsfvxpEfyB3GbICD0UL7V1FZvwBK9YSNdgcEUF24PD6xpNjUXO6QPxR1vbCB1Bj3uxmOMVGyeOcbHfkpwVmT9j7E6NZJXUJtPKfYsaEPjFOBtiDzKRIOjIwN9UcuEr6kgvuqHPOl3Vn0QWmyeLmjfPdhmGCMBqcRQHZUvcsGmhuwOl09SBlKnMnaKe7vvAnHtRjTu6bVV2FpmaPkZwWIzhvSI4mvoXK6SyKD4xPgxojJPJbTtCQVqyTH6vJXvGx2fjEfljuBrzl6Fh1uaNiYHd6D4ijmRQ85IKWhbqz9yMLrNsn5RW5gw9kpTSzGf9Lc6QGT2s2Xb3lS1u4v8kCqHrwEwbxHnzckBK4vHXApmFKJqMtZHVBLuLXl2tEhHmeOyXc0uOOEm7xmTFIGXN85ZHpy2G3oMZHR9slmOD3NSgNnxSgqXQ5Ne8P8LYNpWzZWPsFt2V6EpdE58lw8KsH8cUCuvu6wo2wY0djiKmKXJIHFHf7POkYsm34NlD3ZvAcIH4NHm8Nn16NpRIfp2LMrdvGZyGfp27N0Rw8iYaRoNSlSglmAQqKd0vDXx1EpMNaX20CT0p8LnA1R2rEAnd/b4VDtbPXAFXSv2ozOeEz6yS2H7fLxnLSbNVSRCchBY2BLENQwvMM0Gda6Y0UxbsKfMm5cAp0y8s8B7dJ2whdBDPjakTMTLVkWQhkwdBDyXfdGdZFTZdP66ZLWWjY9fLL6SD7dHoI97gW2NcHFKzoLVEzUvi+YJDkcIr6bSTIs9Xmt3OO7B2OqvRa0cMDwGAzjF8+SEC1QAHaPTxUWrBRg9MbkFHXgnrKDy3RvGNiC8HYXjLSrRr7bV0JnDU+u7xc5DZtJuTbAxfh+hMUFEJKqmxSGp9LHoz4DzUc9qehHuHmB1S7OwjIQX6eVGz4CI40C9Dg7jgFu7U4Ykb4AdiPnrY43fdVg4zItdEDoZsxydWJFODxY7xbOGeb1oNuRhQQe1PSTsgf/hjZt5FhlMXzqmOA+60OMKCr7tq0h63PjDqTvSVJRd6ibOHzuibwb5OubqPV7hakBeujXqLDJa0BdTzPVkKu31tCnJXfZTyt+YcHP3IzWXFJxtVyyfgkFxKa3BE6Ohv0vlCXDO01NhTpz3POHjGYPhyRTqjUxr4SvUF+q4WDDpebDmMAG8oQOCWMPfPQmXIWlP9gEClb9Wbdx1e2X8DECM1QDcYjVJBt9mAW8xu8MW8qmnMUKnpYXUImDPhH3ZLwUNWlm8oU2TZ2DNcTCQ4nqlq5nixStjrs56zVWO1CsoFZaSmRgfYD5DMEEfKcDXYjAVn81RQG7b6dcmahF2XMvNGAxB1gXLAdLPnIc8KXxdxkhRb6HQBxhpHhnsPcqyO5I7jCMOVHfEqsHTUz3P1mFLD8bpujyIfMRjuNNS14h9C8SXrWEHpvaLz4Kc1+BQjNRWBqHbdx81ZZLkjnpJ1L7DHGPLeHgWfTO2HHF7v5EjbowV01Ex5f3hXXAxXYFMspE6TmWL8v6W6MyGQSS5KDb3dWiT7Wt8mRqaPFaKxSjR89+ZXxSkmiSxyXiTnG9k3AGq45fZBvqfebROOwkDh0bg5xV2SHGXSmv7n8Nbv4dqqYv4L/+cdP+CjHrmH+IJvCpEXddmzg8hj2LC9JY4JthdqIsTzF6Dqe8GKhr8qRZxEtVq2MnrmNyxs5BLs+6TXVs3SDnd5Q6D/yn2wVKFoTCJuAAt36b1xDfDMdFO5YDTBgDQR1BdM5mVtntoIevJdiDeYFXLRVCEP1Ls2pBtpTk0lojHjVPk/XAh8EZGXDd7j0vEhKnK7Pr6fSL+YYrF+rSnzrN5T0FcgrQNtAm+P9r0K5vamjDzl0uW0W3T8xXFzhu3UsBiwbamEPQPYZBx3uzbvoTrqF5MxyuDWJ4w13lC3XzIadTYyzZs9LryDC5Bm+mq+3BwNUvnSYAHmLAm8OP67fNx6bQsm6iB6BgjB1GvVG6KxjVFg+NVtCL4YFEp99GJWbdMqeaZ0HjT6nSbXGOy6Al0R24iQnvlbXnsLl9vcFL1hX9OwCwnQEX0uEXQdPwsmrSBGzPzVJg7zbu43SndWp0XgAM2FPwmE4lnD5I+s+0ZQTJAFQLTvBqf2GZFbRbgpXJAhu9Xmmcw5EF7sJFC9LqyQBIsX+3GkK9RjDU7r6sSIdJ3GBv3dBjPX3mRxD7AyRvpb3BHG6GWSaoJjAA13w5LZ8SxQCKuDJwTlklb81NQzJYDIBJ8ndamGzzM/04x7zgzpAiNJyijt3pyUmvQKhuIcJrpUGKUAvPiZIUXe4AuJdLuKGNj7+FQuoxHk6jtRzE+Omq/QP8xt4hKR8Jt8XxPUOT0LQXEVIuw3wWDe3hyLS/uhgyWGqjBF7L/Ff6dUGgXKpmhtUmMUUF0BCNx01tCTqhdyJ2zKQ7w0FO2PmXD7zPQ4uJ+GSDdhnrdJhiZA2MjEG2b7nI01dDhxb/PImmUIpi6p4ZDKGvX/oZ9m4d5lXHh51wLr6qU/7dQhpcbgxWEQ9MImG2dIpiuB/QWMtBoGzbMCGL6hXDWAIc2m29LE+yXi7azXwt+qfEi9RuE5bpGDOc2+wZmIbS7OJ1GeNbMC4ipTa4XrfdLSjJcJCFjTHy3bKYPmRDXmBrJBbBjiS2HWFHEtVgVcfDQq5TIRuZ6B8y5dd6FhVvYhB9yrOVJHH5AeEqSEN/Vzed2Dz7kACQl3UvpLoq7Edrb/R7OINz+9kFNkRpAbvsLbxMu2UHL3p2RDxrI1M7bbLWkvMvXiW0hb/rdP6KQt4rjJRpm6mVNtlm6SBEHw1JQ0wwa6iWpHyRELbZyXlQGYLDsDvKsMiq9sQY8QlRIWeEG8YePa0ogZRqC74RJsyFyhB5J0qxU9rFAm23j6oiI2I70ovU4KqvV3D8pXITgRMwyXrmd+GbmoP0r1YVEpfJ0Hn1JS7JqXRxXsV8LTwWHd15ouAXgvs7xZ7QmTrGbRPfa0HzM5ERrSyZvY9EKmZZzb+4lMa51J8N4v/VrlGB8Mdk66D2aeGVFcb1s7yaZ6OTgf9MWDJ4d0AgDqPH9MWrL/XVezvkkqQ4smxoQOKYjQO9svwPQ6YVWYzR3EyBhcNbbXNFqNkYoE0XPYfaDPLbI6YEzqlA2rTrUZ0sOpGuVd3XT52FwzEmlFbAJvhN2BxTKOuvTjn9PWAHwXeZt8a7vG6BhtbMoTNZm6BkswL1g5DYLybyj+lfQ1oRX8Po8hIuDx7wm3AklmTTGQc6xjUfsJBWZcGfoWYDYiAD0JPrOxN6Gi6tUo88nYC7rVxqb6Y3JN4ZsqFgaq7Swb2eTGr/SmDm5ZjLEl0llc73VsQGCOe7kC5G9rcIgGJAObZwKY8Q1zKVuHLBFYSRxz0OY9HW9tLMBn6tYPDUJnWVGTj5U3g8RoLawdmKcKJoewZRDeNbh7l1T/c8xRw3KUVmO5hHtH71VyxMzw0wFK/nFNbdl6VKnHD7jnBLgjCu/zOpfxUGa1Li53GvfdTLmtcXa6cZN60MaAp4Y0YymGQ+t9mvaoL7a4K5mGlLnOfbPP0kOKUPzEsbXzqSAGvVSuY8g9NpNhcZYWGyK6ZTPCTzHJGTqCFhiYwbxNCZ1dsxnbSJSrFpwUqaL44ZeVNJfA2ePf6r/ELbE7dR3oIBSnbtlmWT2EbgBFq4MNbfyZo5Axon5RwIUMxMn+7POR2bVdoWRGqKNjXmTt6d9Xf5ZgKdSfR4mAZC0Z57yYcK7kz+t5OAPhMk8LwbIzF8rZeIHHIcqJwRKUb7Tkmj1B4IRpB9K2nYIzQK0zWo5bkHz/SfAxVcfM7cv5+eKLGWYQv2oQ8kvI5Ni4fT6VCgPnxaZo7Gt5r8ZnoPsHAfPqb80uexU4ZQsLP3gEJVdzQzBwh1DhlV5QbeIgTQy7NaN3nhdbm6RA5sSZpR3c0OftGJswzDz1XITq/DwJlcOuTSXhU33gWY1v/q9wtLJjxF0SvbvTJgDVQ3iVHpLQx0bCQJKDYoLr+wLSHeW4kN9IxFkCl0iBmr0TfupV7WYaPuUs5WZOWwOqv28lmM2vk4dAwB/nmmVwrWtWk4C3xm3pZLJ3FWCtadvMWh+J+3u0ntYAWR/i8vHrKzhzU3iWmQJGJK2zLJeA04NnbXDBz6YqO5jnGl4d3RUEEb6gxdqnAR7iB8EAJgUnKj86uOqC7mFh5JWWRu+t7TAq3vIm3KnXzGLV7w64wr6CI4kZK2GY7ceCRQTKzbPqVr7XIWo5bQMAVUl3m0CaEZFv3mNoeIQ7x0WIuKKTDLTKNUnhKtJt6FnbfSSVcGBaGJvvoh4Hh6rXIX5lvPqrH3tz0eGvRCu7DuwI6CRUj5A5Z8xbxgYk0yMqI1bLJoOJaIrjxBvhEzg7NchIO4e+HxzMoqnrb9KuArrJUMOowkrlJAtR4S6lpPnx5yj9JOu5BqNGK8qhZ7Zr31SuS5gyAIw3G6Ms3NaL2qRUgnGmGhCG6vPjgL7T6V1q6bNQP8LM1eNtMWdPnfX5tcAiKQGqbbCBKf3JoKxaHAxvzU7LgFqTx4z9Ff0s+AeyTbebR1XxcxrsIsObkIv5WgoKY8nOnxe0oK0wcS+/75jxZoKTyyN5E1WH2u6i5JUuuhuxDqTvSR0HlFbrq8tIP04B5qb2hGAa9HNU4WvFPJcZh5pKnmbTczrtvCBDG6q17VqNDHRqJlHJPA9POf9pHdkz+0Bs20nsm7JJzXQ0nK2dwUGW5EP6EfmHGe7rzHTqxkFZswbKbgzsv1iXJFgFrg7o+Av5lNhVvRRK2Vzjy0uN1i6FFyP7y6k7c9Y2lJt8NfmAV9RKvFqhB7fI6pvuVdyzEmbiJmDm01SMq5Q+aXCKdlvFsnBXrmAGVJiEPB6tLQJrkNDtaGhcLcVI7rYDlaK1vXLRE7qyUMCpJXMGDhWQt7R5bUYIlLRtNuM9hhNHXc0VfbZCAfBs2SYwCqj7YQdMcgnH7GWqcWl3OfUdCY3X3oNSqA0ZL9NmNSD2nHF9NDGxwGbzgEMF/LmACNxXeZM6L5I6ZwKBRi8GFnKeOiDYMeEXZnQvHIVqBCaCNmWMbsN5KdLHVNL+K7ZWUlz9fxlyB1P+F9nYFHYzwqpZ6gP3rL0pORSAeIgzK3IQWDGzvbSHawFc0pAEEMpH/VhxxQF5d4WS0dEQ5o2u2zlS1bBMiCFzLxCbsgdlvI1shnZbuk6P8sv8hw10TMhaIgHeF7XVxrkcYUpmNdv/LR9D2y9CA+2TuTWo5vZR5h7Kwi1fQLtVXK2etz4Ls3H4ylmMBewAhrGSDeHoqsMPgDdNhWWkMg2w84pbGvTXQKB06ttMiODyfszhf4hddvuzR9jUchWDtuEDxjAqEwnKSe28bIvWVcOh1VpzvtIkMR9rjHVvqNxLBnsQR6kqi+DeICLrIXxIlKrmbIPxTKcJIRKRuAEgrH4NLhW3ITqhQyo66UYXUW5pOLAlIKwIvxUSIW18kuu9VMj9O5Ew5k3fXFM+CIEPLqYbXvg7IQbN0rrbn4kIZmuaX4a7ETM9MlfGxe9WW51oAuUBP1XmfAKfBJZC4E+ScGUOf+CtWKoUlo7uBaMa75Y0ysw6SMx2fg0WHNSrmFwmEWFu4x4bJpcBwRHRuZtymuNKeoGmbdsc7hxOBZ2YhPiQ7a1EHWHvtsbjg9UDfx1GpjsDTLhd2ghYRrCwLFGXpN9K6yaPbJ5VSq8JrTJRvmqeMQG1DB6egGlBJVTmnGMUWlEQJNvEEhn00KUuwHu+4AhP4+DR3c6aPCB7EBE47j7MJbHgcfPSzVA/kRjg1aNZdrlzCFJ7qmQ/i2d5wFhBDqaGbDdSw8LbDSLCvAIFbdWgGTP0UYsI1QefHxPEG5D3q6Bg1JLzk7YjG6BRCafOb4f+DbhsBqPrVfyALY8MCuN3kaW5QNwUAjx7gwYmKPbfQrEJJOTwYm9+6WERyJdOw2z4a5QCVxjjTC4ZmPmjmD+Nm5eMtV8cIGufJzM3Qd7t1xJiejSQJ6eFUzl8RTcRXb/3XCbQM6Az2lXEYsVWrk6D2BzIw1gOJ2udqwyLN/S8UE3ggA7nbXWGXHdD7P3p7OVsjnDmHH5b8pKwTaAbnnnoAZZoTLdEHmQzWCzSC/tUz40MT+VXdZqa/4fGz7rZ6iNFD45FtthbsqrNnw3p6gPvRkPwWgehTa9oYozgHRmWFKe2KL5z2g0SbWD+6TkdK1Vzgpl8phOIP14v4Z26mgyFPeeOGqKPnn9FVwANv/ZK0YxjVzB66nDBRVo4hJo9rLqMNq3uXf2UfyI0kd2jbPZj4DqWtyOk3NnMQw8KRHdfNvcvGzCL6mUu5F7jB8D4h7kKEQyY8n2svdrZ6CR6qyrcUTeE4zT17SfmWTLqOT0Nd2Mop5PXJEJKd5UrMtSsoYnBKlCvGdqLgjpz6DcuBU4y6dTbJrKxIrPTeu+eEBuAFC+i01D5NGQ4YZXl2ALHgnV5H3D0U4ahGNK68SuyeUrbxd3nnpKLMgOR2bKVEn+BttqfVTnSP7QZiZbsm49YhPbNhVFOv8qjN2Gq5EF2AnfT6A5GUO8aSQXGyFx+Gt5Fzf4BkapYOd7Zo4dIOedOVWiiPt6GtyZZA88vdhxgUiBXjk8PDrDggUCpRHcUmA2FoAvW8jD5UAQNZ1x1GhhxyWlaZCqLl0Wlcz7nTqGjTOaRiYl+8gBf4y6FMCI2FB6yhA+RJ3K4lwuFr6qRjR/0gy6x1oCSd2ynv1xkwMq57Lh8Z9MP3RiY80s5vI0PJBhg7vzJBL8ORyEC+aOoUsPIg9QKyVJAWDAV3OsD3rvT5Q3z9wF0Y6qB1WW2UYaaH4TDkCI4WB8qMyAxaXE02pV3Gml8D/p6qcDbgtTLT6GNmW5yYHmm19dnrdDRAPRxBKmGpTrpq48oDo0AL8ELAnB5mX0aPPHlkqq85Hk6ViNKdzuD6eRFhtrjvESYB+wJ0qHi7Gel1MXLdMGB1yTqSeBAiqe8d29wfdr0mFXdPgAJNs2Ty7VSO2Wa70j+4eF0jIw7sx0pL3lZt1LZ0AD2SjS5GFqjhDqQmRFXzgubBAoVzc66KG1AXFAgjXT9FRaqJ7oGvFg6xAHp7kU5x4hUjWKHgtSd6L+kyaTY5EjIfgMXYwVoFkFnGhVhpBAUW6ZzdC0UPbdy1PjAE6OJ6ZwL94T0o5AgyV9lNE0x2ZSZQyDZ6cn8TRzKSpkGVPmhSbYxVH8w9LPHIo4YM8J4x6GzGPBbRRdPS91VP9SN9DMA4L6+cNVIuRbhJtrq31kKyMkuxONx73sk73ehTAhoKoAPBc6bqE2n/hgGn0Ca7AoBzGJ6tzBe5IDVtLPvPVPIxFmsOxnHw0ONmmL8Uo0uNWc+YUYwXxPHY18tf5t3ivdz7fXNQ8VjpGr8afXbYqzDJyqb+FV/jjI86y7y6ABvPmrFO7pq8jh4S1wnAQjr4kLmc7rL8DY2DuYQjIlrx7xZB/oeWIRh/j1bwkFeQbTZTVtJcSGxyJl8T8HOyoWlUmCI4fS7Sva5uAZLR34t/LcJ5yDTlVkvYZBeE4OaIkVZeXgGRXTPzH1rFH2SOAYwT0XxKxs6PMI1BvJ7b/rR1NJPDqhlCFafhvBhlk0t0S/wf2Nq7Jmfg8cSN2DLSVXxCPwybsoOoU8AuFO97I0k5+w35IBM7mBTH39SwV4jpHHMV4mWBYDmLBXnwGaK0xbHBbcMHgzD9SQLw5nHf2TnjxLE4PGA6F1GqXnfe8fA7mp+F/dNz4aF+9FPzwq0Cla/A8hJ3RP9TTVkjgaEzlLK/ZVhIBr9xO4NI9U06+iQ5KQ3AdZHjbeV7Y7/9w0edMeJ0mVAmKYnjpSqLRVDgAXRCoLIhyUjNlGxkhvpDMGk41uQG1UHIkE5zAhQJWS3zMfLUmNNYjhVYKHpYgWe3tVvTUJNTiMEPlUiQ3NpOaROR03vL5DRAH5fqo1YlmuqQWpKhZnIOhbCp44w/jVLIrp/w/5p87EQPIEWuSfmSsYTPWkmGQDrdSxVNbm/bgd+zBSs4UHGtk3HJEsZKh2jLEsI+vYPRCyMLSUNhjKkdgQdrqZKxNRRsAp6PE20Dq3ozAkIzHXS2NbLJ8W6qdg9zrGc6Lc8VCr6Gz0ivnRpFWqL2KsHE+zVu0G9TTKWIHyKa4Mn7Fl+OBsb1apyEJfRQU3G08SlnYKYcKp3CxXMmh7otszj0J8Df9tYctq9W3bF9jxRDr3kdRyYtBRQdQv0O6J3dJZDrbxmvRgxNzH2m8+6GP5cI4BWvs3+cOZZ1ZVbqgpzmW5VTzRDSC6aIuYOSf4YOcWZ6dRc9bcsMUyv9aOdhBBxfc3TRBhWhzm7OR2X8CA4vGUe4dgvT7+qoG5ZFJCJg5DAWa2Dq6kUdFxSH4J1a4+zHFRhRQJnO9kR8x39VNLmcUw/6q7yDs2xsp3qDXQ5tb+PMcD0nhGfkGkc8UjDXQnhM9GXUmuMXAeUAnw+EHG1NTHtIiMOCCp0i0Zxtgx3o99dFw4njKTHYGL8rvTUrWx/RjVdQF2TE5SZuhABoBBKSjBmVfmn3S7hxUvsMTOPX71xzsbjkw5ZhSiEp/xbYlm0YnBqF0Q1YH2A0PXrZKUlqbaSAKpq1QUqB3H0TO/03EoAu8xzAOC7AA03v3JBj85pdaxyHaabV4a1ScBzufntkoxZYEUO8RnDmRbiOKAvVeeAZ8Q0zrWc8IUwANVC0r6tQCvpbMYiBFO7ZyYaMfy0HOzrgHy0ZUxLY2YKdEpw7RrmsMjuRNd85kyDpY6ZiTqRjJBqbQxEAA9c3fhIzyeHHjSrLSlm6sA9v1V/PFbqOm5M+0s4e6GF44nLt41eGFaUDRrvtoL1Pnd8gjNBHc6GS8+eqWobWS1V3dS6EWa7Zjpfqc9mmdiBHIgJq9RB1m4m2kHOoE6YbJAJsVp9cFpv1XI8xsYx11yBq8rsEHKwTsLVhRfLZf8wCNphMdrRfgd9cGmn+Mcxj7jrAJCbbVTw1t9WE/4pl3qkKnKBZ20n/JStF6g/wj5NI03ujH0x1Vbs6ScpJnHGOPJGgGi9PtPXbBi2VCSAf2DxhMYgXNWLVjUNeujPyAQ/XcMS5OmvutKYX0u0xtGYEQ3tdxVYyA6SwzY1k3Ec7lLXy7H47zTrH3C8zaljuqGLE3gSJkmZjrCZqm1yUK7WZmmR8cdQ7g2OOGHwlKjfFj5t1nCvtYdEEbHTe5H6fBfoTxoCOwq0TS6Eqw3ObYQr4AG4bqpYLaCvODq8dQ4Jh88N+9D4emNxIxf7f0s6WbNBuOVN+c2Mzqc/kKcVYFcNDa8OLy1OSSGBF4C1ycu3S1WYhSAdq1TgaT9XGSIj5j7NbqMpbgX7qfhRVc5XBebfaUGGQPqtB7MUNsQVuDQaGGNU4qEwmVvqrBvRn3v/WW1Sbv2uNtIIAbnwv3xnKtI+ezJfEhnz7+DWDE5gO5GPxvz8eKIRMJjGeDF63oKIeGOLUjUFXnLTP7LXqiRXN6dDWfnnxg+x/hgnhCCuQpzhZu1lALg/m7VwqDbLiuJHrI1p2C0JdadNDlVXdUwI3Ogp6dxbYoUBXl8L2wX1GmQfN7+0YtbbnWeFCzmzq1ve7egzqQaFX+EWsjhMIeL4p8HJjNhC5wMGw3tamRXK6ONvKqlAZMM0LAHnQfY7B5qKNaT6h9sEm05aDC+zEhpH59wrbC2ZJzLwGJJ3HqZ5WwZQfSbhU5nXzNTUw0bZOgvpvPL1IAu5A8j75Z8SWbexQE4SMG7NqqGgeOjwzy4HuA7cdMvC3Y3Kom8E1c5vtNQoPRtRKmLzLeZiEYqDjkm+qqL3NWR8k4KyXzyTojNmR8Cwuh/mnMe1LKLRp2ODNFmEV3PeGBR6I4VEtd7/sPnkN9gX+FlERvy32mm6qvBL8a8tzMqkBHWBfHb0Xmr4FnbyvbZsH/Lmr2HLwW9JXPkL1KiIt7hqKL12FZCn+bBeX8qAuaJvU79Xs1DQyD5PILFQzLiCeHL7Wh18/f9h32i/EoScBZjRZadTD6shYMX56XtpREnB0E2HAYPvwnzv18lcSKJPeMHEjwzyKz2SjVK0B0KxeCV6CuWpD8FYFxyD06MF7l4P6nvmpezAlcHn6awrdhFNwdymm7R4tSlLVLTGP0D+Rp+0GbEqIczgy4p2J0iqzbIJzmjnKFXdZE9E6Bkhwx8PS11lGBl0PrSJvhhUgqTe4VD4UYp7Ij1eKzQCyRDIXvbWrn4T7GjnbWIu3WKdfTc77EiqpKOTtbHUJsu4iPptLl3Pn7GbUMTRYCfMlOlXGpH4cBh0jaeJ0sGYgTOhqAUiVLQMsIfv24dtMzPzvgmEeGEc3ldFzOe39DPl4GHRhNiJIkvdBhSjAqFRXcdPvCiDiXhZCnBCi52VAHQ6ajlqMd1XjjwILJqRrxZSGebdETGNfVgjB8g2ldBSzx6fbDkUZOqvQ5uj1tJiNq08vxIXBUiV9pLkzdNCyZ9Rvj3jcmu8+VtxaqcnrJg5BmaoXwJrmHapzdtKs7RPpgcuN7UZJINVSVV1SHMebxZX5l14FY0YEwZ27VTZ0qeHRpyfFKxZjkz94y1gYYdH2PNAw73tNhqW2nvRa80j3GdJoemJJo54vmqeqeHe0yFiPX8XSQBePO+tHZu9vStoIS5gDYpfxMpumnrHijaN86BLjIIZ3wvtIw9AWP2l3iw4xLUwvY3CkGpEm5HY0m5aOHP+4BsL60PM3pdl4sTLIoYR6pUdhQXfBWlLbJ34nhHqTP+xtm3Bkr5QO6aI5lrrMOtgNH8ut1TtbhUJR7P3vsa4Ynb0fjX2uHhKQ8q54cnS30jdxz8RdJZyMqmXn8+bqGH5uO51vFvMqr4MQWlMcVPPSfRWn5Yr4MVuIyjuMBX27YeyYwQdq0w7Rpc5lTEyMXgoqlSWEmu7w/jqcKFJPVWYTxiSfGgyIzGoBoMLbs0C+2D+6XdyaJaxFNqXd1zE8gTHDu+X9YVgWmDTjLphKy17mvLjY16W5IVvMuyRF0KqyDKKbNyTYXVB+RSpWr7sqyJC+qTAbX8+ZwFu82KlMBAOomUPFCOOj34d1/2hM+TxEfNmvoWHiXnLAsVPqXbMNk2dEjKVRZzfKzmjZVH9ff7VY2yUSgRv9nd3X+qT9PsjLY1ykPP0FVV6LfvGCCJjFrcjO6aKYjubKpeTt2iqmeDeJhSyoMiu5NxuBRyvCuTdXVuUgAoTEbYX+iUZJgNIC5d8cbDtajNI3+Hqg0bu4r1jRyjOs8fsAC3HCRf8hw5wsX/ACQ4yOd4cz1VpqeXPM6LPVoTSrcOjr2G6bAEZO6vJUaxpOqWw8N5tgAuAVO1PcDwyPHk7TcLKTPvIBgErgL8O3cPMI9XpXgcKBCcAHvdPGtpNyMOdZXnAQLL2gRe5L0izrx7fS49Oq10jSG48CQkjb95hDC/Sc8Q1VyOwW9AoyEhi9dCDR53Gy6mpHglKUNm1P3XkzcrxSMrjEX8Kqy4WCe4l0Ew2ZbtIU7mBvqBxdD4uXzg15YIuhzSd3hyN3C4/lQhCihELQc7SKFe6LjWS72G0KVB6kNK9TqbBKHPeBA4yAjAtqnqm4dG2WgU9Q8NbfS4iSp0IY1GVRnRRIw2cXTSH+OG4FKPV3JSGSkYpd8Bx4mHlZUwDNHc+dZaYNF0NOhmxXNdfD4wfm4dZ3q5DmgCwTz+nDPiInXcROLYsLqc2DLzsvZx9M4vVV92gnb3xdLDnkOgVrBGlT1iTW+PRjJZhRH8C2tEz85Xf5sQXTnVvZVK98uwI6E6iXIDkGx4lzybedNxmRGs5j64tYbS8QBcuhaLSfK5jMwk7egbKmLmh+HUjTjqYFqGQN2hfSeMeyVaSnlLWOwFM1v9jbJ3XGMhKZgoJWkcpAo9RxZmbeOkpvVMkb7h/CD63DxyR4P6NaKKSWa/7+zrQjSuOBvxRhnSkAJApwNpCjzv3wQnO0BSOYMmBAKjpeutfe2vQMNkAkpzFrUvfnqUdnnyudSZMdvn3Umoc4ZKIcWdwRMgHUuFtsQfMIldTB15rdPAyP6OuMZ0tnNgLJR40HU9PwdTGtuBMkojNm4HBLaZPg0IiqfTDPAKCyGmmg9CdNrR5d0dJ5E6pRvdFQZsm4F81HgAfvYG0ZN1nPn3pzOkqjm7Tn4HRhq0wqO6ZDVaBiQd9Qeamx2atwqi7VqnLR/bAypKTpg0DgGw+tSw7gTuDKoKcKwz76JZimSPmFoYgLKpoY3UzqHtwcwJSoIDzB29WkHPUs6RNqOvMNKQdr0FGLjAqd5yt9vyuSqzaHZnC4uHnojRCFINVSiCD6vQV0vE7b8DNomjgoNtJwcCQz7qmgudR2sU8kdiRU1rZMbAP+0QmXJKXyvMB5Tnx95ffDW8t3RWfVYcBJKu6eMkTNkuEBiMVNxE5L2U/RcwajmL7ETIgNYVB3GbXg8e29rbdiK6l2R5tfnlbWMIOd4rwbb6dM0xQuBvtkiq0B8oDqUCzObBmTIcX0fjEGcoaH2tO8IyyB45E0mnjMfdbHtJ3JhAKmbx5VH/UwiX1d0KgwV4V1BeKeASQpiOSePPrJnYQBeoHVMg4YDJD3tTIrgcK6nL+pRDZP70eT2rqKvd4Bls5gdCtrE/5nDUKt3jvmxj9VJB0CMm9VX0WRDz5RuVSNrfbmwfPr1oHSPqp4AKVL8BhmHWwIfMQhI5aDFevp8s1pIJ+RW3P42tgLYQVOkkPSfy2IYPDtETL8OgCcgbvXDpBF8lrCRqqFdt5EYfKzMQQJQldmwfpHvtVeXmkgpNCWHUpCv8OBBLVI6ZIsuCqEcC5hrbw9rAqyZmmI6ltGz1mbtk1RbqBFxKdpLKAfwlNTXy0bYwSEBR77ohEKAK03jtWQwEHnWUwBqmgp9b1iFkJ1flZbdq9EeRIGKqPaH2yOTYTZkn6utmfJ3oa02dJNV6S5zpr7Z7DxcT3m3FyAZqNeJBvJCHHOZ8m8wk3T5sv1Kmw1AStaw4hhMqYPObz5hsBGUQ1rbgx5mbY1j2YKeu2Vb1RHjfxQVaUdb+Ha3HBKMPS5nm5qs9vaK7BTafGvYY0zlkv6Ebxc3HRcd8tz2WbaMVlxhxgjlXbE2ug0ExdcKrJxl5K1kGpgeWZzmRzX0hWCptPg1EeZNPHuoHj5F7E9KzSCOAP1bT9Fam3RVpjoVdbBvAk3Fj7uIqZMfhVdjuu0x2x2s2dF3C0DShHW2KgClKwl9RAC+I9i56rWTTQLBiTShDQfT1QzB9KNyLfgkAxhMs0oEMebtvlBe2YwHm9CAQrJL6NHP1ex3d5ecD0RpemI31+8eGwxaS0HxCzT6PpK7rphCgwgBxR5ZCCqe80SYsVKi4ytyRF7QZfBC6bb3Ie3xUYdeRe33jT7ysTlVuVLtd1O+QjR4J5qw1+aI33jr2jX2HmR3SiWbY5kTlhQhzk27XTgB7JuRTsoB6X0qJbjkFIHnmdMSmEMUXTj25ZtIbdYRyYXz8MfGRSRmpx9yHcIT9UhI/Lco5lFB39C6LP2xLQqB4nSNSydMrO2rZGbhKrmf0Pt6KBlCxWgLA6N54e1FGnAjEx3ScRdo6PBJsP3BSY+yx9Zwdz2wAlo+DHNpNSqXt+qpF6Kiq7Vrm5hGUp+YIY4dr9oD1VhR9z1WfGXnHpdbg4HhB7ohNa3+uBPuGTakc49HyZumW7QLHbMBzmALj3ev6SaJs5kHHixuYX7X9IgGyNUbqXRCmkv6KaCwQ7kTRPr5AZDQlrfbqY7mQz5TV3u11UmEJl+AtdR1ZMcp7ZMqSmRXcbdxXk7q3tlKsL35sUxf2IlPQO0WgXdJK6sQZcAaJe1CthB6nUEWz32pqi07PBOtXhnakyzJxnVmkYkmjUwPwiXwzsNGqZj8vbfBWvAfUdRkIATvRVM8CpyBHtaHB2gr93bPADs2eMZWBr7fyU3PDnWOT55Nkm2dTtz5Zgj7CETWp4n6MrYCnbD1B92dCwqlHc5QEGH9Sog4DHt1i5fvO8Vb6x7yl8AKaelazWP5WJ3+8EXPLm4dwPtoL16xeopXMv0L6YdtSTBa9T7Y1yOyIprvKx3SxP0SKW1mg0H3FxmgrMb0YxHZRx9z1aCCgfwf2XYg5c4XZCBrRJfKvq0D89u8qgSF9PHjDTOEXpw01jWXDg5eZI5sFbYnH02aNBhsSw2hjbl+MuBuANjeogFnCPW+CZjmZSXF6BJ4glBrEPZwWxSCVL4c0VJZ3dUsT51gUxhhvx6m3nuGmwTdz5O3dhXwQf3u9kTqJrJQiJu1xqkpTsRU0hw0SSKMZMVz1pnBI0Q8klDtYfgmc8L7GM4gC0INdXoMm4gX60qUtcw8gjG9vRXwRXP1RRswDw4oclaEkB2pOs4w28JQ9U7Q9JeJLPfhwMkaDt6E7NUO9fE6hTI2hHu3jpXY3k9BxEpq4R4UZ2WAz5m3E1MYzWt7bT6Sk3Q8DT7R0YZLmTSDH5PWK7ijqkVVpIhWdSMaw8z3fDO2YdxpANQqaBvwWPUlsUGeqZW5GQJq3rQtHvC4SQgxp3aGVbJlgzmV5jhwUbswFLrsk8HR2XnN7ltYY7AnIpz8CobSO3I+pqnr76kH5ryBnZ9AaJE3Oo3KOmdYYfnYWRRmXUsqAyN8mhHqyaTq5LkDkqY0JxtW1d2yMDk+2nVk7nMY3KGKtVK5rxNCMwBZK84jo6sTG8FZzvMkIlt1SKjVN6ZsT7TrGfJv3FiknM7LNjUyYh23SBGbwrvZtE0sSYSjwqRzRnVkXzyR6ymsc9E0AmsnfR1w/hjk3UATxhjEsaYi8BZMolBFVAuTUM6s1iplSOfiWAfsElEGkXECXwQeOKaMj8kUxt
ARCrJNBVdyGNMKgfRpNYIlFpN8pYIjDFJ4JPAbuAU8B1jzAkR+byIvGu82G5gUEROAs8D/90YM5iv

mFRuckkETstJIp0glU4tUVRKqXzJ2/BRAGPMk8CT1zz22UmfG+B3xj/UMpBIJ0ibNHZr5l8NEcnUClJRf
JZviaJTSuVDoTuL1TITTUZx2V2IyKxldS6BUiuDJgJ1lUgygtvmzqmsy66JQKmVQBOBukokGcFln7l/YI
LWCJRaGTQRqKvkMnR0giYCpVYGTQTqKpoIlFp9NBGoq+Sy4NwEp82p6w0ptQLklAhE5Psi8g4R0cS
xwuWyztAErREotTLk+sL+D8CHgXMi8hcisiWPMakCiqRm37h+gtNyEk3p7GKlrnc5JQJjzLPGmF8EbgV
agWdF5BUR+VURceQzQLW05tJH4LTrekNKrQQ5N/WISAXwUeDXgEPA/ySTGJ7JS2SqICKJ3JuGnJY
mAqVWgpyWmBCRHwBbgG8DP2+M6R4/9JiI7M9XcGrpRZIRHLbcKnlOm1P3LVZqBch1raF/Gl836DI
RcRljYsaYXXmISxVIOBnOvY9AE4FSK0KuTUNfyPLYq4sZiFoewonZN66foKOGlFoZZqwRiEgtme0lP
SJyCzCxElkx4M1zbKoAIskIpa7SnMq6bC76kyt769Czw2f58qEv0zLSwtbyrXzq1k+xtnjt7CcqdR2ZrUbwN
uARMruL/TXwpfGP3wH+IL+hqULQmcVXvNr1Kh/7yceo9dXy8R0fp9hZzIef/DAnB08WOjSlFtWMNQJ
jzLeAb4nI+4wxjy9RTKqAdNG5jLaxNn73pd/lN276DbaUZ6bNNBQ1UO2t5ree+y2+/67vU+IqKXCUSi2
OGWsEIvKR8U83iMjvXPuxBPGpJRZOhnMfPrpC9y02xvBHP/sjHt7w8OUkMGFX7S52VO7gS/u/VKDo
lFp8szUNTWw95QeKsnyoFSaSjOCyVneN4JlLzzASHeEt69+S9fh7m9/L8+3Pc2743BJHplR+zNY09NXx/
/9kacJRhTaXtYZWYo0gbdL83aG/432b3oc1zdJaXoeXh9Y/xFePfpVH7n9kiSNUavHluujcX4lIsYg4ROSnI
tI/qdlIrSC5bFw/wW1zE02urLWGXmh/AbvYubHixhnL3d9wP690vkJXsGtpAlMqj3KdR/BWY8wY8E4ya
w01A/89X0Gpwll2ncXpNHQfgeOPw/HvQ/dRMCZvt/vnk//Mg+sfnHXPZq/Dy91r7uZ7Z7+Xt1iUWiq5ziy
eKPcO4LvGmNFcNjdX15+51Aictszqo8aYnDa7n5PICOz9R9j/DXB4oHR9ZhbL4AWwbPDQn8C298Ai3r
dtrI2W4RY+sfMTOZW/r+E+/vbA3/KbN/8mdivXPyWllp9cf3t/JCKngQjwX0SkClhZbQIKYwyxVCznRG
C37FhYJNKJnPsVcnLyCfjxp6HuJnjL56B03eQgofcYPPsncOYpePdXIMe1kWbz/XPf5641d+Gwcrtevb+e
MncZe7v38ob6NyxKDEoVQq7LUH8GuAfYZYxJACHg3fkMTC29aCqKw3JM20majcvuWrz1hlJJeOoz8J
PPwBv/O9z721cnAcjUAGp3wtu/CMOX4NFfhHRqwbc2xvDkxSe5q+6uOZ13e+3t/Mf5/1jw/ZUqpLnsOH
YD8AER+WXg/cBb8xOSKpRIMoLb7p7TOW6be3H6CVIJ+N5HoX0v/NwjUL115vIONzzwGQj1wdN/tO
DbHx04iiUW64rWzV54kjtq7+DFjheJpWILjkGpQsl11NC3ySw18Qbg9vEPXXV0hZnL8hIT3PZFSATpN
HzvYxDsgzd/Flw5TlGx7PDG34UTP8w0Ey3AM63PcGv1rXPu6yhxlbCuaB17u/Yu6P5KFVKufQS7gG3G
5HG4hiq4uaw8OmFRRg498//BcCu85U/m3t7vKoJ7Pgn/8duw7i7wlM359sYYnm17ll/b8WtzPhfg5uqb2d26
m/vX3j+v85UqtFybho4DtfkMRBXefGoELpuLcHIBfQTHvgcnfwgP/P78O31rd0LDHfDTz8/r9AujF4ilYn
NuFppwS/Ut7OncQ2oR+iqUKoRcE0ElcFJEdovIExMf+QxMLb25zCGYsKDZxYPn4cn/lmneybU5aDo3f
QhO/AD6z8z51Jc6XmJH5Y55D4Gt9FRS7Czm+ODxeZ2vVKHl2jT0uXwGoZaHJa0RpNPwg1+HHf8Jyp
vmfv613MWw/X2ZjuNf/O6cTn2x40XuWXPPgm6/vXI7L7a/yE1VNy3oOkoVQq7DR18kM6PYMf75PuB
gHuNSBTDfRBBJzKNG8PrXIBGBG94x93Ons+XnoOtQZiZyjsKJMCcHT7K1fJZRSrPYUbmDlzpeWtA1
lCqUXEcNfRz4HvDV8YfqgR/mKSZVIJFkJOf9iifMq2ko0Asv/A+4+7/CHOYszMrmzMw2fuEvcj7lQO8B
Gosb5zxs9lobSzfSEexgMDK4oOsoVQi5/hX+V+BeYAzAGHMOqM5XUKowIskITvscE4E1j0Tw7Oeg+U
EoycOWj5vfBm17c+4reLXr1Sl7DsyH3bKztXwrr3brVt7q+pNrIogZY+ITX4iIHdChpCvMfGsEc5pZ3HM
Mzu2GHR+YY3Q5srth88Ow9x9yKv5K1ytsrVhYs9CELeVbeLVLE4G6/uSaCF4UkT8gs4n9Q8B3AZ1Xv
8KEEqE5rxk0587iZ/8EdrwfnN45RjcHW96eWa00PDRjsZHoCF2hLjYUb1iU226r2Mbe7r3odBt1vck1EX
wG6AeOAb8OPAksfF6/WlbmNaFsLmsNtb+eqRFsenge0c2BpwzW3gUH/3nGYgd6D7C5bPOirRxa660lZ
VK0jrUuyvWUWiq5jhpKk+kc/k1jzPuNMf+ks4xXnnByfjOLc64RPP/nmdrAIq0WOqPNb4P9/zszTHUar/
W8RnNp86LdUkTYWr6VfT37Fu2aSi2F2TavFxH5nIgMAGeAM+O7k302l4uLyMMickZEWkTkMzOUe5
+IGBHR9YsKKJKY5/DRXDqLuw5D7wnY+OD8gpuryi2ZUUQXX5y2yP6e/WwpW3hH8WSbyzZrP4G6
7sxWI/h/yYwWut0YU26MKQfuBO4Vkf93phNFxAZ8BXg7sA34kIhsy1KuCPht4LV5xK8W0XxqBG6bO
7emoZcegW3vWpraAGSWq970Vtj3T1kPB+IB2gPtbCjZsKi3vaH8Bg70HtB+AnVdmS0R/BLwIWPMxYk
HjDEXgI8AvzzLuXcALcaYC+Mjjh4l+x4Gfwr8JbrRTcGFk+E5LzHhsufQNDTcCq17YNPb5h/cfDQ+ABd
egtDAlENH+o/QVNK06DuLVXoqcdqcXBi9sKjXVSqfZksEDmPMlL8iY0w/MNtbu3qgfdLXHeOPXSYit

wJrjTE/nulCIvIJEdkvIvv7+/tnua2ar3AijNs29/0IZk0Ee/8Rmt+S2XJyKTm9sO5OOPrYlEMHew/SVLoIS1t
ksalsE/t79ufl2krlw2yJID7PY7MSEQv4a+DTs5U1xnzNGLPLGLOrqqpqIbdVM5j3EhMz9RFEx+Dw/80s/1
AIG98MB781ZcP7A70H2FS6KS+33FS2idd6tKVTXT9mSwQ3ichYlo8AsGOWczuByVNHG8Yfm1AEb
AdeEJFW4C7gCe0wLpz57FDmss+y1tDRxzJ7D/sLNBG9ZjvEAtB9+PJDiXSCU0On2Fi6MS+33FK2hYO
9B7WfQF03ZkwExhibMaY4y0eRMWa2pqF9wCYRaRQRJ/BB4PLS1caYUWNMpTFmgzFmA7AXeJcx
RuvUBRJOzq9paNoagTGZxeU2v30RopsnsTJ9BYf/7+WHzgydocpThdeRn0ltVZ4qDIaOQEderq/UYlvcnr
JJjDFJEfkksBuwAd8wxpwQkc8D+40xup/BMhNNRufcWeywHCTTSZLp5NSO10uvQDIGtbNVHucvkkh
xrGOEUz0BukaiBKMJ0sbgtNvwOW2U+Zw0urbxxlOPEL7vjykt8nO473DeagOQmU+wuWwzB/oOsLY4
D+spKbXI8pYIAIwxT5KZhTz5saxzEIwxD+QzFjWzRCpB2qSxy9x+JUTk8r7FRc5rNpfZ9/XMEM55bvg
yk+FwnB8c6mRPywD1JR7WVXjZUuPH67QjAsm0IRJPEYwlOT7moylVzl9/8W/oqHkzjrqXuXPt5kWPa
bKNpRvZ37Of9zS/J6/3UWox5DURqOtHOBnGbXfPa5cutz0zl+CqRBAZhnPPwHu/Ov2J85A2hp8c7+Hxg
53saCjhE/c1UeyefW6Cu/hNfC5whMeaPsQ/XTrB+UsbsN3ez70bK+e9M9lMNpVu4hvHv7Ho11UqHxZx
MXh1PYskI3js8xvemXUI6dHvQMOuzM5hi2QkkuALPz7FC2f6+eW71/PQ1pqckgDAaM2dlPW8QnP5M
C5Hkvds38b3DnTwyNNnCcaSixbjhIaiBgajgwxFZ174TqnlQBOBAuY3h2CC254lERz81qIuJ3GmN8BnHj
9Khc/JL929nkr/3Poy0g4/wYqd9F38N9b4GllX7uVX72nEZsEf/uAYvaOLO5/REovm0mYO9R1a1OsqlQ+
aCBQwv1nFE1y2a1Yg7Tmemc1bt3NRYnvmZC9f3H2Gt22v5YEt1VjzbMoZqbuHSwMvUetZB4DdEt62r
ZZb1pXy2SdO0Dk8j72XZ9BU0sTBXt3RVS1/mggUsPAawVVDSA//KzQ9sOBtKOOpNF998QL/cbSLX7
57PZuri2Y/aQbBips4QYgG6+rr3L6+nAe2VPH5H51a1JpBc2kzB3oPLNr1lMoXTQQKmN+CcxOuqhGkk
nDsu9D0pgXF0x+M8cdPHKc3EOGj9zRS4ZtfbJMlBc45HGwdujjl2M76Eu5truALT54kEEks+F6QqRG0j
LQQS8UW5XpK5YsmAgXMb1OaCU6b80ofwYUXwFcJJQ3zjuVQ2wh/8INjNFf5+YVbGnDZF+fXtDX
WTYW9mJqeV8i20+pt68rYVF3El545Qyo1/T4GuXLZXTT4GzgxcGLB11IqnzQRKGARawRHH4UNb5z
XdVLG8Ni+dv7Xiy38ws313NW0uEM7z0YvscZVj7FseIezb27/pi3VJNKGx/a3Zz0+VxtLN2qHsVr2NBEo
IFMjmOt+xROcNiehRAjiYTi7Gza8Ye73jyf5q5+c4VD7MB97QyPrKnzzimUmZyOXWOOsIlS+g9LO57O
WsQTefVM9L57t53jn6ILv2VTapP0EatnTRKCATI1gvonAbXMTTATh7FNQuTmzZ/AcDEfi/PETJ7Fb8
OE71+F35WfzmrPRNhqc1QTLt1Pcvx9JZe8Y9jltvGNHHV95voVQfGFzDJpLmznaf1QXoFPLmiYCBUA
oEVrQqKFQIpSZRDbH2sBwOM7nnjjBxiofb99eh22BI42mM5QcI2ZilNtKSDn8RP3rKOmdfqnojVV+mq
p8/MurlxZ033J3OU6bk0tjC7uOUvmkiUABEIwHFzSzOBAZhtaXYd1dc7hnki/8+BTbaku4b1NVXpZ6mH
AueokGZ83lewQrdlLW8dyM57z5hmoOtg1zqntsQfduLm3mcP/hBV1DqXzSRKAACCQCc96LYILH7iE0
0gq128Hpz+mcVNrw10+fpb7Uw73NFfO671xM9A9MCJc04wz34gx1TXuO227joa01fG3PBZILGEXUW
NKoE8vUsqaJQAHjTUPzTARuu5tQoAPW3ZvzOY/uayOWTPPQtpq81gQmnI62Uu+YtDmO2AhWbJ+1
VnBDXTE+p42fHO+e972bS5s53Hd43ucrlW+aCBQw9z6CRNhw7rEI+/8sQOzxKpIBJ6y9M6dzj3SM8N
K5Ad5905p5LxcxF/F0gvZ479WJAAhW3Exp9x4kPf0EMgEe2lrLDw93MRye3+6sDUUN9IR7GI0tfBSSU
vmgiUABmUSQax9BMmI49X/CiBiafsGN093PLWc/TSg++/nBWJJ/fOE8P79zDT7X0qyCfjHWRZW9DK
d19WikhLucuLea4hk6jQFKh528M76WJ3/UO+von+hgmq6XYwydTGDSmbJ2y05TSRPHBo4t7BtRKk80
EShgatNQIpS+/EI2mUkZzn0ngqdSqL7did0tlHr20F16lqd+kCSd5RyArvY0P/lhkm/+c4AtRRU0Vi7ePAGT
hv42i86zNuJZRoSejV6i3pl9z+RA5a2Utz897bXjByC+R6j3uqk+6+XIo8GszwvAwLE4J74eItSZouP5OOce
jZBOZso2ljTqxDK1bOnGNArITCjz2D2kE5kmn8ClFDa3sP5hF+U3Zt5JG2NofTJKOm6ofsP4nINkFOdIG
2erRrl7+A5efTHFvW+6+tfq0GspDr2epmxdnMFYlNoLaxhcE6eiPksHrDEQDBNuH2Ow3YYrOUaldGCZ
FNht4HaD3wfFfij2k/IWcXxvMbGI4PIaWg872PFgjOKKKy/WpyIXWe+qy/59l2yivP1pXIE2YkXrrjqWPA
/JM4LzboO4hFBxlPRJw8Uf2Wn8+as38Rk+naDtqThrH3LhLrMwaUPnC3Fan4zS9C4PzaXN7OncM58fjV
J5p4lAAVc2rm9/NkY6CZs+6CYykKbt6RgDRxKU3+hg6HiC2Kih4UEnYo2/CPadxlFUS8IEuelO4WfPG
ErLUtx4s434wCgvPxmirdfL1the+n/SxRsdQsJdxaln38jNYz+kyAxBOg3RGASCMDJGZ/XdXFj/DkrjHUQ

ddbRLDTvjT+NMhSCRhEgUwhFMKMKJtR9G6GHT0MuIx81I6UaO/fgWdvlewFXmwJQUYQLn2VK3
HodnDGwW6ZRFfMiPJQlcvlGiZhu1+79P75YPYux2Ui4XiZSL2M9cOG83yPjKGxtq/fy0v5eScw7szwpr3
+JCRBg5l+TCv0dpeHMmCQCIJdS9wcmlJ2MMHE2wcetGvnr0q9n3dlaqwPQ3UpFMJ0mkEhCyM3A4T
OO73IgleKttbPh5i5GzKQYOJ3CVC9V3OLDskzp4e45ilW7AETyFsSe47YYxXnvGw9F/7yNqPDjtKbY7
X6fHZhitX0N5TTEOY6iPtnPM9m5u872I054AhwO8HjpijbQN1tHc0IPLaceYMXoHyzgYej+33HARp+P
KTN+WtlriAR9NtR0QuwsTi1MSjRELd3I8uItbux4nERzmI4Eo1amXsBIpRryNnGn8MM5kgKTNTSRVQ
W37XvzBi5S5voKkDMRTHN3y69SFuqk4dZpEsZ9YZTmRNTXcU1nFTyNdvPPMWoJtKZzFFmMXU9
Q/4MRTeXVLq80prHmjk0tPRdla4eSeDi9n//KPKeocIxkKMVbUiHfrFjb8pwdxVOR/CK1S05Hrber7rl27z
P79+wsdxooyGhvlbY+/jc9YjxDqSVF7Zw5LTSSTyMgI/PQr2Ip30XLuJNv6HVg2O/b1awlWN+OtKKK0
OMFYNM5PjvVy18aKq1YS7RkoJRj2sHNzK05HkkvdVXT1VbBxbfdVL/gTZceCPrZvuoTHFaO1q4a+w
VI2ruvCbru6ickYaO2qweeOEq5+nhOR87yh6GaCI7UMdW6luOoCLu8YxkBkrIbQSB3rix7DWR5jpPY+x
s7XkAy58K/pwR6L4AiGsY8GcA2N4OrrJ54ydK9bR8mOO0mWV+FvdGB3Zxn9FE9gnbnI2MFh+uLrW
dfzTbzr3Pjq386x/lowaZLxNGUDJ7jvrWVUfvzXEEu77VR+iMgBY8yubMe0RqAub0ozcChB7b1OiESxzl
7Cau/G6htERgMQiUEsjiQSEE+CMRi3AxxuTPUAUa8NeWgXFVWZtvgSABKAYe/5IRqrfFOWk66pGE
HE8PqxzViWweVM0Ly2C4cjNSXG2soRnPYUh041kTaC3xOlae3UJAAgAutq+zjfUcdIbCvVpTaGujYT
GqmjtO4MDmfkcjlvSS92Z5i2vv8HX+AC4Z51IIJ/XT/YbCSdfpJFfqirJgBgQIZHGT16noY9T+EaHsHU15
Cqr4GSosx409EgVkcPVmcvpqaS0sZ6THmCFu9/waRslIXsbNlhp7LKRjJpeP3V2zj9/LM0nvo09Y98EbHr
n6VaWvobpwgmglTH1pKOpyj68RPYT53H1FWRqqogXVsFG9dj3E5w2jF2O9jtmY7b889mZhKXrqNz4
DA7i6bWJFr6gsSSadaVeaccE4GailEqSgOkUhZOR5KZphWUlwYoLQmQTltZE8BkNpthY0MPL3emKe9
9kIQ7Qtmak9jsUxeRc3oClK89jtU7jN2bIF1TNn0cAqa8hOQt2/jhUJif31KJs38QGRyBrr5MGY+L9I3NJB
+8G1yZ56QEg1R28GLby/zmrZ+4fDm7Xdi23c0x81bWdHyVni/8GXWf++MZvzelFpsmAkUoEeK2o+sp6
TwOdU7iv/Tuyy9g00olYbQTGu8HwCk24umr38lHEikOXBrh1vWlmXfK07Db0rO+sE+wBKwcy8YlTLd
vD1uKIjPeH8Cy0jjKUpSMvkA/72G2E2pK3PQEYhzpC3Hb+jWwbs2s8RS5/KQlxmhslBJXyeXHS8ss3G6
L8Dt/g/T/+VPGnnmG4oceyuE7VGpxaIOkIrH7OWq71+PdXkH6tu2zJwGAsXZwl4A9U9YuNmLpyTNvD
XvPD1Jf6qYoT8tKz6Y3MUSZrWjWJDAh4S7HYMMdbMup/Na6Ilr6gvSNRWYvDAhClaeKzmDHlGP1
DTbOX7Io/9Vfpffzf0o6FMotaKUWgSaCVS52/jyuv3+MQNE6PA1z2KFs8AL4ay5/6bDsxMyVZpfzfUFG
I3GaqnJbhC4fehIDlNiL53ROtLgR/8CRnMo6bRbb6kp46dwAseTUfo1sKjyVtI9N3f2sutqitzcF6zfjbG5m8F
vfmlPcSi2EJoJVzBhD9x/+EV33PkjKPYbNnuMIsnQSRtuhqPbyQw6xE0tl1uwZiyTY3zrC9vqSJVlLaDo9i
cFMjWAO4p4qBIM7y7v2bCqLnFT6XfysZRCTZR/ka1V5q2gLTk0EdrtQUWnR0ZGk5D3vYehb/0w6HJ5
T7ErNlyaCVSz4/POkRobprW4GzxwWRBvtBHcR2K/UIOyWjWg6TjJteOFMPxur/XnbaSwXMRMjko5Sb
Jt7jSRS1Ih/4HDO5TfVFBGIJjjVNfu+BWXuUkKJEJHk1Bf5qkqLtktJHLW1uDdvZuQHP5xD1ErNnyaCV
coYQ//ff5nid72byEgx4gnkfvJgC/ivXrLBIXai6Th7zvbjc9tpKJ3fJjeLpSc+SJmteMZRSNOJe6sRk8Kd5Z17
NpbAjvoSjnWMztpfIFhUeSrpCEytcVRW2ejuTpFOG/xvehMjjz029+CVmgdNBKtU5NBhUiPDuG66mVS
gGIcvtw7PzGihdii6OhE4LRtdYyFC8SRba3PvoM2X7sQgpXPsH5gsUtxEUX/ui8R5nDa21Zfw4tkBwrPsc
1zpqaJtbGqHtMsleL1Cf18a19atpEZHiJ48OefYlZorTQSr1PCjj+K//35GRsHYozhdOb5yj7aDu/jyaKHL1wu
mCMSj3NRQWtB+gQk9iQHK59g/MFnck9nNzDPWmvM5lX4na0rdvHSmn3R6+iGu1d4qLgWyj0wqr7D
R0ZlELAvvHXcy+qMfzylupeZDE8EqlI5ECD73HL6776G/P03SNThlrf5pDbZMqQ1cGgozOJbA4xEctsL/
SkXSUaLpGEXz6B+46jolzfgHDiEm920qm6r8pDEcbBuZtky5u5zR2CjR5NQ1sysqhK7OzAgk7x13MPbk
k7PugaDUQhX+r1YtueCLL+FsasJWWkpvb4qoYwCnLYe5A6k4jHaA/8poofahCBf7Q2woLyLFzE0iS6U7
MUi5rWRe/QOTxd3lpG0uPCMtczrvxvoSLg6EaBvKPhfAEhtV3iras8xXKCuzGB1NE4sZHGvXAhA7c2b
uwSs1B5oIVqGxp3fjueUWAPr7UoQdPdhzqREMtYK34nKzUMdIhJa+IJtrivA4nMRn2PJxKXXH+ymzz7
9ZaLJwSTP+waNIOvck57RZ7Kgv4dWWQYKx7M9JlaeK1tFLUx63LKGszKKnO4WI4LlpJ4Hnnp93/Erl

QhPBKmMSCUJ7XsZ7661Eo4Zo1BCSfpxWDjWCwXOX5w50j0Q41xNkc60fl8PCjo2EWQY1ApNJBO
WOktnL5iDlLCbpLsU3eGJO55V4Hayv9PLS2ez9BTXeatrGpiYCgPIKi86uzHPp3nkTwec1Eaj80kSwykS
OHMFeXYWttJT+vhSlpRbxdBTHbE1D8RCE+qGolr5AjFM9QTbV+nE7bEBm+GjCJMlhTlVejaaDGMA
nizd8NVLcjG/kNLYsY/9nsr4isx3nkY6RKcfK3OUEEkFCyanNRxUV1uV+AvfmzcRaWkgF5jC8V6k5yms
iEJGHReSMiLSIyGeyHP8dETkpIkdF5Kcisj6f8SgI7tmDe9uNAPT1pSguERKpBE7bLOsPDraAv4bBSIrjn
WNsqvbjGU8CkFlJ1CYWiQL3E3TH+6lwlCzq8NWU3U3MX09R/8E5n3vjmhLO9k5dj8gSixpvDZeyNA/
5/UIiYRgbSyNOJ65Nmwi/9tq841dqNnlLBCJiA74CvB3YBnxIRLZdU+wQsMsYsxP4HvBX+YpHZQT3v
Iz7xkwi6O1N4StOYbPsyGy/CgNnCXlqOdI+SlOVD6/LNqWIHXvB+wk6En1U2BenWWiySFEjrlA3zsjA
nM5z2jPrEe05N0A8dfV6RDXealpHL045R0SorLTR2THePHTDDYReeXX+wSs1i3zWCO4AWowxF4wx
ceBR4N2TCxhjnjfGTNS39wINeYxn1UsFAsQvXMC1cSOplGFwMI3bH599xFCon3Qiyr4eWFfupcidvfbg
EAdxU7hEkDJp+hNDlNkWPxEYy0a4ZCPFva/P+dzKIidlPievXxi66vFaXx0Xp5mnUFll0d6eSRyuG24g9
Prc76tUrvKZCOqByXP0O8Yfm85/Bp7KdkBEPiEi+0Vkf39//yKGuLqEDxzA1dyMOJ0MDqbx+YSURGft
KE71naY9XU51iYcy3/Sji5yWjZiJT3s83/qTQ/htXpx52hw+5lsDJolndG7DSQE21xTRMxbl0qQhpX6nH0s
s+rPUMioqLPr6UiQSBuf69SS6ukgODy8ofqWmsyw6i0XkI8Au4IvZjhtjvmaM2WWM2VVVVbW0wa0g4
ddex7V5EwC9PUnKyiyiqdiMNYJ0KkG6v4WQq46a4pmXqS5001BHrJfyPNQGJguVbaW4/yC21NwSns0
Stq8pYe/5QSLjS1AIQp2vlosjF6aUdzgyw0g7O1KI3Y5rUzORQ4cX41tQaop8JoJOYO2krxvGH7uKiLwF+
EPgXcaYWB7jWfXC+/fj2rQZgO7uNGVlFpFkBNcMieD8meOELC81lWWzXt9p2YkWsEbQkeij0l6a13u
uknEXEPDX4+/fP+dwSr4P6Ug+vnL+yZHWNr5bzo+ezlq+qtrhwMZNYXU1NhA8emH/gSs0gn4lgH7BJR
BpFxAl8EHhicgERuQX4Kpkk0JfHWFa9dDRK7Nw5nBs3kk4b+vpSlJVnEsF0NYIzPWOUBVugNLfBX
HaxEytQjSCYihBNxylZ4LISuYiUNOMKduEM98753MYqP2ORBOf7ggDUemrpCfUQS019D1RTY6O7
K0U8bnBt2kxk39yTj1K5yFsiMMYkgU8Cu4FTwHeMMSdE5PMi8q7xYl8E/MB3ReSwiDwxzeXUAkWP
H8fR0IDlctHfn8brE5xOIZKM4MjSR9A3FuF8WwelVpiUtzqneziwX7Nd5dLpTPRSZS9dklVPjWUjXLaF
kp5XkHRuO5NNsAS215dwoHWYYCyJ3WanyltF62jrlLJOp1BeYXHxYhJnUxPRM2cw8cLVuNTKldc+
AmPMk8aYzcaYjcaYPxt/7LPGmCfGP3+LMabGGHPz+Me7Zr6imq/IkSO4mpoA6OxIUlGRGf4ZSUZwX
bOSaDie5IWzA9zh6yHuX0Oui/Y4xF6wzuL2WA+VjtmbrxZL3FNFyllE0cDc5xb43XbWlXv5WcsAxhjqf
HWcGzmXtWxDvY1TpxKI2429pobombMLDV2pKZZFZ7HKv/CBgzjHE0F7e4rKisyLeyQRuWrUUDqd
5oUzfawrcVAeaSPuy31Eb2bf4qVPBEmTpD85TEWeO4qvFS7dgnusFWe4Z87nbqj0EUukONMzRoO/ngu
jF0hnWeW0otIiETf09qZxNTUROZLbfspKzYUmglXAGJOpETQ3EwymCYXSlJZlfvSRZBiX7cpooInlk2
9w9JBwlZOyu3O+T6Gahrri/ZTYirBbUye55VPachAu20ZJ98+w5jiKCMmsUnq4fZRE0oHf7qMjyz7JIsL6
DTaOHonjbGzUDmOVF5oIVoFkby8mmcRWWUlbW5KqahuWNV4jSEZwjieCjuEQFwZC3Fhfgn/kNDH
/2pkuO4XDchArwISytnh3pn+gAOKeChLuSop7XpnzuV6njaYqHy+3DLDGX8+ZoezLTdfX2xgeTjNWuZ
nosWMLDVmpKTQRrAKRo0czE8lEuHA+SU31lR97JBXFaXMRiiX5WcsgO+qLKYp0A5Bwl87pPg6xE
1/iGkHapOmI91PlKF/S+04WKdmEIz6Kd3ju+wasLfeCgUi4lLNDZ8m2ap9lCY1Ndo62lZAcGCQ1MrLwo
JWaRBPBKhA5chTn+vUEAmnGAmkqKq/82KPJTB/BnnP9rC3zUup14hs6SdS/bs73cYqd+BKvQNqbHM
Jrc+POZRntPDGWRbB8J0UDh3FEB+d8/ra6Ylp6UtgsR9ZN7QHq6y3GxgzBzfcSOT63JbGVmo0mglUgc
uQwzsZGzp5NUFd3pVkIIJaKc7Y7TDJtaKz04YgNY4+PEPfWznDF7EQEm9iIs3TNQ5di3VQXqFlospTD
Q6hsK6WdL8551rHbaWNTtZ9ouJQTg9k3q7csoWmjnXMV9xM+enQxQlbqMk0EK5xJp4mdOo19QxPnz
iZpaLjSoRpNRrGwcaYnxPY1maWbfYMniPnXYua5z6NLHETSU/fizQtjaIt1U13AZqHJ4t5qEp5qSrpehD
nuM7ymzIObGk4MnM46egigrs4iavnpOjplgr5SC6KJYIWLX7yIVVRE+4ALr0/w+6/8yEeiAVIpO1vrinE5
LGzxIK5QJ9E5dhJP5rIcRNJLs1JIT3IQl+XAay3eJjQLFS5pRtIJiuexd8H2uhqSSRcHurL3NViWsG6tcDow
/5+PUtloIljhIseO4Wxs5MSJBOvWXT288tnT7TgsN1Xji8n5h04Q89VjFjAM0ylOokvUYdwa66JmmdQGL
hMIVezAFWjFk2WvgZk47RY1nnr2XDpAMsv2lgANG/2MeuoZPNWe9bhS86GJYIWLHD1KoG47kbCh

etJooWOdowyERyl1ewGwJSO4A61Ei+beSTyZU+xL0jSUNmkuxbqpcVTm/V5zlbYcBCtvorjv9TlvZLOpt
JG0vY+nT2TvNLbZhep0F0ef0hnGavFoIljhokeOcp4trFtvR8bb/YfDcXYf76Gp2oHTytQGfEPHiXlrSc+2S
c0sHLI0TUPdiX68Njcea+alsQsl5fATKt9GaecL2OJT9yWejt3mpMJdw8mhU1wcyH5eTUmE8+eTpFPZa
w1KzZUmghXMxOOMtA/RG/BQX5/5UafShscPdLK9vgSxRXHYXNhSETyjF4gWbVjwPV1LlAhaou3U
2pdfbWCyuKeKaNE6yjt/ipXKfSTVGu96PEVt/PuhTiKJqYvaldaX40wEaT+tG9WoxaGJYAWLnj1HV9N
D1NfbsNsztYHnTvdhtwk31BYRSYVxWi58g+O1AfvC3107rfyPGkqYBF2JAWodFXm9z2KIFq0j4SymtO
sFZJrRQNcqc1WRljhVZRH+40jXlOP2ulpKBk5xbt/cl8FWKhtNBCtY6MhROktvpmFtpvO3pS/I8Y5R7m6
sQASiyRBuI5naQHHjotzTKQ4ied5f6EKskwp7KY48bUm52MKlN4BJUZLrMhQi1HnWY/e2MhCIs6/16r
2OLZ+P8mgHFw/1kUpq85BaOE0EK1jL/l78zgQ+n8VYJMG/H+7i3uZK3M7Mjz2ailASuETMt2bBfQMT
3JaLSJZNVhbT2Wgb9cuwk3haAqHyHdhjIxT35rYJfZ1nA13hi9y50c+LZ/vpGI5cddxbW4rXmaTr7EgeAla
rjSaCFezCcBn1a4RU2vC9Ax1srfVTU3Kl+SeaDFIS7CZavGHR7ukSB3GTJJVjM8hcDSVHiKZilC+D2cR
zYSwbgcqbcYV7KO7bN2t5h81JpaeOvvgZ7mqs4Lv72hkJXxmW66ipoTTVR8sh3dhPLZwmghVquLWfg
L2CuuZSdp/owWYJW9dcWa8/bdLEUxHwryNtORbvxgIey0k4HZm97DycilyiwVWd6145y4qx7IxV3YIr1
E1Jz6uz9hk0eJs4N3aE+jIX2+qL+fbeS4xFMp3O9jVrKOo+TuvRzOY2Si2EJoIV6uSTJ6hIdXO0e4yWviD
3NFcwaYkhUmPn8BiI+3Pbj3guPJaLUB4SQczEaYt3U+/IbevM5chYDsaqbsMeH6Ws/afYUtN3rPscJfjsxb
QGznBDbRHNVUV842cX6RgO46itxd55FhEY7AwRS6Z47nQvf/nUaT73xHF+crybVFoThMqNJoIVyBh
Dy+kIPleQZ0/1cf+Wapy2ST9qA7auF/HYPBhr8X8FnJaTYB4SwdlIG9WOcpyLWYMpAGPZCFTcTNrho
aL1R7izbEgzYa2vmdMjB0gbw7Y1RexaX86j+9r5zqEeov5SklaYr/zLUW7/wrN86emz9AdjxJKGLz19ll/8+
muMhpd+fwh1/bk+hl2oORnsDBKPJjkUinLPLRWUeq7+MfsGjtApKZz2/Gzt6BYnodTiJoKUSXMqcoFbf
Dcs6nULRjLrEiVc5RT1vo535AyBiptIeK7uBC91VmFZdjpC51nnb2ZtuYfaknpaB0L0+8pJDLVRkt7Cn39
oBxX+K/0/79xRxzdfbeXX/+UA3/7Pd+Cw6Xs+NT397ViBDu/poHjgNJWbNlBfdvWCbJKKUdb5Uwb89b
jytIa/W5wEU+FFvebFeCdFNh9FNu+iXrfQEu5yxmrvJuksoazzBSouPYlntOXK1pcCG3ybOT78GunxvgC
HTdhU46dm0wa2Ri7iCqUosV29PpRlCR+9ewOReJKvvnh+qb8tdZ3RRLDCjITjHN3TjjfWxca1U4dYlrc/
S9zXQMACt+RneQav5SaQyn1ZhdmkTZojobNscNYt2jWXEyNC1N/ASN0biPnX4hm9QNWFxylvewbf8
CkqxY1NbLSFrl5fKF27BnvbeTzVbsZaA1Oua1nCx+5t5J/2XKR9aHETs1pZNBGsIKORBL/55b240imKq
qe+c3YF2/GMnCFQeQuBVCRvu3p5bR7GUsFFu96FeCcey0mZo3jRrrksSWZZimDlTkbq3kjcV4sz1E1F+
zPcPNbPyb7ncQ2dRMbnaRifD5xOfN4Eoy1jWS9ZXezmLVur+ZtndJE6NT3tI1ghBoIxPvL117g7ZlGV6I
C6+qsLpFNUXPwRgarbSNucjKWCbHQ35CUWtzhJkiJuEjhlYR27KZPmcPAMO7zNixTd9cFYNuKeKu
KeKgA8iSC+cAsX+l/m7os/Iu6tJlyymVhVBcXBS1wcXotJG8SaOq727dvr+PR3j3C+P8jGKv9SfyvqOqA1
ghWgpS/Ie77yM7avKaZuLE1Z+0HS1ySCso7nSDmLiPo3kDJpIuk4HnHnJyABv+VldBFqBScjFyiyeSmxr
+4XsJTDz8aiLbzuhEtN7yRcuhlXuIvi9Dkq9z+GR0bg+GuQTk451+ey89ZtNfzjC9pXoLLTRHCd+8nxbt7/
v17hnTvreHhNBZJM4nbEMd4rTUOusUv4ho4xVn07CARSQXyWO6+Tsrw2z4ITQSQd5UTkPJs8C9sjYa
XwWR7qnVXsD58h5qtnrPp2xrY/TGzUQZmvn8Tre9n13dtpeuX3KOl+GdJXVi59aFsNu0/00Du2RNuIquu
KJoLrVDCW5PceP8rnnjjJpx/awv2bqxk6NUyJ6Sddf2UrQysRoerCDxmruZu0LVMDGE2F8Fp5qg2M81se
hhPZ261z9XrwOGudNXmP9XrS5KynOz5Id2J8wxuPG+P34/K7OZ+8n4t3fB5jc7Jh3+e57fF7WHfgf+Aeu
0CR28G9zZV885XWgsavlidNBNcZYww/Od7Dg196gf5AlD9773aaq/2YtGH41AhlXYdIN6yfKEzlhceJFK
0j5ltz+RqjqQBeW35fXIstH/3J+a+X3xbrZiA5wgbXmtkLryI2y8ZWTyOvBA6TMOPNQLVV+LrOkUwII/
FqBte/g4t3fJ5Lt/wuznAP2596P9t2f4iPlh3nsb0XCcenNh+p1U0TwXXkeOcoH/zaXv78yVN8/L4mPn7fRrz
OTH9/oC2I3QHe3gukazPDLEs7nseWihGsvPmq6wwmRymy+fIaa4nNz0hqjPQ8Fp8Lp6PsDR5lu6cZm+iv
6LUqHaWU24t5PXgMAFNXg7RcpKQ6Rf+lK/MJ4v56+jZ9iHP3/R2B6l3sOPdlnrJ9iqPffwQS+VkLSl2f9
K/sOtA+FOa3/u0gv/yN17mxvpg/f+8Oblxz9azgwWODlFoDpNZtAJsNX/8R/EPHGKl7A1zzYjqYGKHYlt/
OV5tlwyMeRlJTx7fPJG3SvDC2n3Wu2lXfQTyTza719CQGuRDrhIoyiEQp9YzRe9HGtWvQGcvOaN29tO

76Yy42/wqesz/E/M12eOkRiIwUJH61vGgiWMZGwwk+/x8neMff7cFls3jk/Tfx0NZabNcMEUxGkoxeCFB
+8VVS6zfiGW2hrPNZhurfdLlfYEIkHSVJCm+eJpNNVmr30x2fw+btBl4NHsWGxQZn/ezlVzGbZeMm72
ZeDx5nKDUGDXV420+SSgjB4WlGAYhQtO4m/tXzSxy94VNw6RX4nzvhyd+FgZYljV8tL5oIlqFU2vAv
e1t54JHnaR8K85fv28n7bluLx2nLWn7g2BD+ShuO3nYcZULFhX9ntO4BUs6pawkNJEYosflhCZZxrrKXc
SnenVthAwfDpxhMjrDds3FJ4rve+W1etnoaeW70dcINVVgnzlJWl6L73MzTg+7eWMG3zwjpez8F7/xbiAz
B/34Ivv4WeO2rMHSBKdUKtaLphLJl5njnKL/3+FGMgd97+AbWV8zclm9Shv6DAzSYM8jaGiovPs5o3X
3EPdl38OpM9FNmX5oZuuX2Eo5FWoiko3hmGvljDPtCJ+hM9HOrdys2K3vCU1PVOMqJpeP8JHaC9yK
USzfnWhtoujWBfZq5fJuq/ew518/+1iHuaKyCW38FbvowdB2Elmfgxb8CywZ1N0H1NqjcDBUbobwJfFVc
l5tBqBlpIlgmIvEUX3r6DI8f7OADt6/ljZuqkBz+4IZOD2N3W5Tu2U1pc5DRujcS99ZkL2ygM97LTd4tixx
9dpYIdY4KTkUucqtva9YyCZPk5cAhgukIt3m34dAkMGfrXLWkSXO4rosbD76Kf/P76G6xsXZrKmt5EeH
+zVX862tt3Lq+HLslYHPA2jszH8ZAsBcGW2CsA44fh0AXjHVmjlVuhrqdmURRuwOqtoJzZS0GuNpoIlg
G9pzr5/e/f4zGSh9/8Qs7KfbktixDOpWm6+UemiIv4fRECDS/iYSrfNryw6kxjDH4Lc+0ZRbbetcaXgseZ4t7
Az7b1fftTQzys8BhyuzF3ObdiqXvNOdtg2sNPRssks9fJLHzNO0ntrFmUwrbNH/hG6v87Gsd5qnj3fz8zmu
G6IpAUW3m41rRURi+BMMX4fSP4JUvw0gb+KsyNYaSdVBcB/4a8JaDtzJTiyiqBU+Z1iaWKU0EBdQ7
FuVPf3SSfa1DfPSeDdy8tmxO5/e90k5J8hK+I68Rf+PdpGdIAgBnopeoc1Ytafu7x3KxwbWGnwZe5w3+W
yix+elPDnEqcpH+xDBbPRuocswct8pNrbeWwOYInlefo/fGMs4eK2brLdlrWCLC226s5ZuvtHLr+jLqS3J8c
+AuGa8N7LzyWDoFgZ5MrSE0AKOd0HcKYmMQC0BkOPO4SUPpeqjclGlyqrkx81HWCHnYIEnlThNB
AYxFE3z9pQt885VWHtxaw1/8wk7cjjk2iRx9hcHXDVv6nyfd2ISpmKY5aFw0HaM11sld/p0zlsuHDc46bA
jPjb5G1CQosnmpc1Sy2b8em74ALKr0xg1Ud4wxEH6dttNv5XDRHh5cfyPVWZJtuc/JA5ureGT3Gb7wnu3
4nPN8ObBsUFKf+ZhJPJhJGKMdmQ7piy/C0MVMwqjaAjXbM4mh6obM10V1WoNYIpLPja9F5GHgfwI
24OvGmL+45rgL+GfgNmAQ+IAxpnWma+7atcvs378/PwHnWUtfkEf3tfHd/R3cvLaU991aT1XR3Gb4ekZ
bKPnZ13n+1DtYkzpBUcdZzJvuhVl2oHo5cAiDYZNb1+1Z6WzBCJUvHef8PQ8wHNrAj278MpWlbu713
8xtvq2U2ouuKr/7RA+DwRifeftWitwFeG8YD8JQK4y0ZpLEaAeMtmcmvZU3ZTqqKzZBRfN4p/XGTLO
TJok5EZEDxphdWY/lKxGIiA04CzwEdAD7gA8ZY05OKvObwE5jzG+IyAeB9xpjPjDTdZd7IjDGEE2kG
YnE6RmNcqE/xJGOEV4+N8BIJMG9Gyt4y9YaqotzTwBWIkhp10tUnnuc9tYSXh/7AGsCRykbPIW5705
wzb6vwPeHfspNvs34lrB/QBWOcyhI2Wun6dz6BobsWxnddoSjZa9wPtZBhaOU7Z6NbHavp8lVT5W9jJf
ODnC8a4xfvnsDdzaVY1sOL7LxYKaZaawLAp0Q7Mt8PtqRSQKl6zMfZRugdG2mBuGvyfRJeMvBXapN
TpMUKhHcDXzOGPO28a9/H8AY8z8mldk9XuZVEbEDPUCVmSGohSSCrpEIPz3dB8ZgyIzXT6UNiZQ
hnkwTTaaIxFMEokkC0QSjkQQjkcz/gUiCUDz7KIyZ1JW4WV/hZWtdMRur/FOa5x2JABu6n8SWjmNLx
bCnIjiTY7hjA/jDHZQFznG04wEOm/cRdKzBloxS3/USTneMZH0VZFl/PpvDqTaqxI9DWwNXDXsiSXlb
gKRVRUf9/SAWrlQr3Rt3c3jNID1mlDGurEbqN25sMSfOlIMiy0Wxw8X6Ej8lbid2sbAhgGAJCIKFZP6Xi
c/BJtakx8iUR5Dxc678tmY+a3SVcbtvLXNiTKbvIdADwR4I9UN4AMKDEB6GdGKaJ8QFDl9mhJPDBw
432N1gc2WO2V2Z0VO2if+dmf8te+Z/sWWawcQan60vk/rbBBh/2TLj/5j0pI9rXtImrmFNXHPS/xOPXy4r
4/cS2PIOKJq5GXg6MyWCfL4q1APtk77uAO6crowxJikio0AFcNV0VBH5BPCJ8S+DIjJ4bZlc2Ioq62y+0o
WtYmZM2pjxn2wmYZnJBzGYyQ91AQdmuFyJFbGttw3O+JZ+JPIacftJhDTWNevNB2IBilxF05w5KbJl8
AYv3wLRAEXu2Z+L1eCq58KA8C+kLQcGQdLdjPqvvFimBZJy1S/ykrEMNIbjeVn4SAQZjhhHtU+wBB
sEybRAX7+6g6ajK2B653n6+ukOXBdvD40xXwO+NvG1iOyfLrOtNiKyfyA4oM8FmediUJ8LQJ+LCSKy
v2Msveqfh9nkswGtE5hc52sYfyxrmfGmoRKu95StlFLXmXwmgn3AJhFpFBEn8EHgiWvKPAH8yvjn7we
em6l/QCml1OLLW9PQeJv/J4HdZIaPfsMYc0JEPg/sN8Y8Afxv4Nsi0gIMkUkWufja7EVWDX0urtDn4gp9
LjL0echBXucRKKWUWv50kK1SSq1ymgiUUmqVu+4TgYh8WkSMiGRfgH8VEJEvishpETkqIj8QkdJCx
7SURORhETkjIi0i8plCx1MoIrJWRJ4XkZMickJEfrvQMRWaiNhE5JCI/KjQsSxn13UiEJG1wFuBtkLHU
mDPANuNMTvJLOvx+wWOZ8mML2XyFeDtwDbgQyKyrbBRFUwS+LQxZhtwF/BfV/FzMeG3gVOFD

mK5u64TAfA3wO9SmEmRy4Yx5mljzMSU471k5mysFncALcaYC8aYOPAo8O4Cx1QQxphuY8zB8c8D
ZF4AV+3mzyLSALwD+HqhY1nurttEICLvBjqNMUcKHcsy8zHgqUIHsYSyLWWyal/8JojIBuAW4LUCh
1JIf0vmjWK6wHEse8t6iQkReRbIsk0Sfwj8AZlmoVVhpufCGPPv42X+kEzzwL8uZWxqeRERP/A48CljzF
ih4ykEEXkn0GeMOSAiDxQ4nGVvWScCY8xbsj0uIjuARuDI+L6+DcBBEbnDGNOzhCEumemeiwki8lH
gncCDq2x2di5LmawaIuIgkwT+1Rjz/ULHU0D3Au8SkZ8D3ECxiPyLMeYjBY5rWVoRE8pEpBXYZYyZ
84qkK8H4BkB/DdxvjOkvdDxLaXyNqrPAg2QSwD7gw8aYEwUNrAAk867oW8CQMeZTBQ5n2RivEfw
3Y8w7CxzKsnXd9hGoq3wZKAKeEZHDIvK/Ch3QUhnvJJ9YyuQU8J3VmATG3Qv8EvDm8d+Dw+PviJ
Wa0YqoESillJo/rREopdQqp4lAKaVWOU0ESim1ymkiUEqpVU4TgVJKrXKaCNSKJiK1IvKoiJwXkQMi
8qSIbJ7ntb4pIu8f//zrEwu6icgfXFPuD8dX/zw6PoTzzoV/J0rlz7KeWazUQoxPsPoB8C1jzAfHH7sJqCEzCQ
0RsU9asC9nxphfm/TlHwB/Pn69u8nM8L7VGBMbXx7ducDvY14xKpUrrRGolexNQMIYc3mC3fgihTYR
2SMiTwAnx9es/6KI7Bt/F//rkEkkIvLl8b0OngWqJ64jIi+IyC4R+QvAM/7O/1+BOmDAGBMbv9+AMaZr/J
zbReQVETkiIq+LSJGIuEXk/4jIsfF18980XvajIvKEiDwH/FREfCLyjfHzDo0vuqjUotAagVrJtgMHpjl2K5k
9HC6KyCeAUWPM7SLiAn4mIk+TWb1zC5l9DmqAk8A3Jl/EGPMZEfmkMeZmuLzg22dF5CzwLPCYM
eZFEXECjwEfMMbsE5FiIEJmvXxjjNkhIjcAT09quroV2GmMGRKRPweeM8Z8bHzjoddF5FljTGgxnii1u
mkiUKvV68aYi+OfvxXYOdH+D5QAm4A3Av9mjEkBXePvzmdkjAmKyG3AfWRqJI+N75p2AOg2xuwb
LzcGICJvAP5+/LHTInIJmEgEzxhjhibF+C4R+W/jX7uBdeimK2oRaCJQK9kJ4P3THJv8TlqA3zLG7J5cY
L7r9IwnjheAF0TkGPArTF8zmcm1Mb7PGHNmPjEpNRPtI1Ar2XOAa7zpBwAR2Unm3fpku4H/Mr6EMy
KyWUR8wEvAB8b7EOrIvMPPJjHp3C0ismnSsZuBS8AZoE5Ebh8vVzS+cuoe4Bcn7kvmXX62F/vdwG+
Nd4AjIrfk+BwoNSutEagVyxhjROS9wN+KyO8BUaAV+OE1Rb8ObCCzp4UA/cB7yIw4ejOZvoE24NVpb
vU14KiIHCSzHPjfj7fjJ4EW4BPGmLiIfGD8mIdM/8BbgH8A/nG85pAEPjo+2ujae/wpmR23joqIBVwkMz
pJqQXT1UeVUmqV06YhpZRa5TQRKKXUKqeJQCmlVjlNBEoptcppIlBKqVVVOE4FSSq1ymgiUUmqV+
/8BmgJ4JfoAnXAAAAAASUVORK5CYII=\n",
      "text/plain": [
       "<Figure size 432x288 with 1 Axes>"
      ]
     },
     "metadata": {
      "needs_background": "light"
     },
     "output_type": "display_data"
    }
   ],
   "source": [
    "sns.kdeplot(df['CreditScore'], shade=True)\n",
    "sns.kdeplot(df['Age'], shade=True)\n",
    "sns.kdeplot(df['Balance'], shade=True)\n",
    "sns.kdeplot(df['EstimatedSalary'], shade=True)\n",
    "sns.kdeplot(df['Tenure'], shade=True)"
   ]
  },
  {
   "cell_type": "code",

```
  "execution_count": 17,
  "id": "7244c370",
  "metadata": {
   "execution": {
    "iopub.execute_input": "2021-12-21T16:51:28.450803Z",
    "iopub.status.busy": "2021-12-21T16:51:28.449763Z",
    "iopub.status.idle": "2021-12-21T16:51:28.639792Z",
    "shell.execute_reply": "2021-12-21T16:51:28.639088Z",
    "shell.execute_reply.started": "2021-12-21T16:22:23.044838Z"
   },
   "papermill": {
    "duration": 0.228366,
    "end_time": "2021-12-21T16:51:28.639937",
    "exception": false,
    "start_time": "2021-12-21T16:51:28.411571",
    "status": "completed"
   },
   "tags": []
  },
  "outputs": [
   {
    "name": "stderr",
    "output_type": "stream",
    "text": [
     "/opt/conda/lib/python3.7/site-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.\n",
     "  FutureWarning\n"
    ]
   },
   {
    "data": {
     "text/plain": [
      "<AxesSubplot:xlabel='Exited', ylabel='count'>"
     ]
    },
    "execution_count": 17,
    "metadata": {},
    "output_type": "execute_result"
   },
   {
    "data": {
```

"image/png":
"iVBORw0KGgoAAAANSUhEUgAAAYsAAAEGCAYAAACUzrmNAAAAOXRFWHRTb2Z0d2FyZQ
BNYXRwbG90bGliIHZlcnNpb24zLjUuMCwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy8/fFQqAAAACX
BIWXMAAAsTAAALEwEAmpwYAAAUx0lEQVR4nO3df5Bd5X3f8fcHMP4dJGCjEkmumFglwWmNy
Q6QupNJrUb8aGMxGUNx66IStcpMBonrRvcdqoEwtSeuKXg1nTUICw8KT9MQlFTaqrKdj1pzY/FEMy
PUDbYGGkAbZDA2NSkYr794z5rX8SuzrW8Z3fFvl8zd+453/Occ56dkf3hPOe556SqkCTpUI5a6A5IkhY/
w0KS1MmwkCR1MiwkSZ0MC0lSp2MWugN9OPHEE2vf1ZVYzNte12GxZo1a5i
YmFjobkjSESXJk7NtcxhKktTJsJAkdTIsJEmdDAtJUifDQpLUqdewSPJrSR5O8OMpnk
5iTHtrZvbOuTbfuaoeN8tNUfS3J2n32WJL1Wb2GRZCXwj4DxqvoJ4GjgIuDjwFVV9U5gP7Cp7bIJ2N/qV
7V2JDm17fcu4BzgU0m17fcu4BzgU0mO7qvfkqTX6nsY6hjgjgzCPA28D4PA28D7i1bd8OnN+WN7R12vZ1SdLqN1XVy1
X1NWASOKPnfkuShvQWFlW1B/i/gE/wfFUdaM12B/gE8A0GIfEEB/wfFUdaM12Ayvb8krgqbbbvgdb+hOH6DPt8V5LNSSa
STExNc39HyRJS1hvv+BOspcPG2M+zjQEzKmlDHr+xnSO/t+bqBfD/B+ljJ/3ea0aPJ8nsWczLiPcn5fK6Ax
7V2JDm17fcu4BzgU0mO7qvfkqTX6nsY6hjgzUmOAd4CPA28D7i1bd8OnN+WN7R12vZ1SdLqN1XVy1
X1NWASOKPnfkuShvQWFlW1B/gE8A0GIfECcB/wfFUdaM12Ayvb8krgqbbbvgdb+hOH6DPt8V5LNSSa
STExNTc39HyRJS1hvv+BOspzBVcHJwPPAZxkMI/WiqrYCWwHx8d/4Dc6/eRHbviB+6TXn/t+5+KF7
oK0IPochvobwNeqjsSBVdHpmnVH+VWqqaL8nOw1a8Vnonx5KnaA4EAFz7F/xB4Ze9n/p/kbwneqaggqAWxuuVbkvRavT/T51tqq2AFsOKj/BDLOZquo
7wAWzHOdK4Mo576AkaST+gluS1MmwkCR1MiwkSZ0MC0lSJ8NCktJ8TJsJAkdTIsJEmdDAtJUifDQpL
UybCQJHUyLCRJnQwLSVInw0KS1MmwkCR1MiwkSZ0MC0lSJ8NCktSpt7BIckqSB4Y+30zmeTJrkkwmeTDJBz4bdkfN3Bk9m
Z5PH2vby1T5JrkkwmeTDJ6UPH2tjaP55k4+xnlST1obewqKrHqjoN+qjqqjoN+EngJeA24DA24DJgV1WtBXaBz
gbXtsxm4FiDJ8QxezXomg9exkpOmg/9exbpkOGEnS/JivYah4Zm/Hzm/LG4AbauWZsk4CzgZ
1Vta+q9gM7rt96nddb8DLCiLa6bmzLZ3rzZLVZ/lSSbk0wkRuYah1wFZPAB7qcSq8Q7u8Az4xtv
7WCQ5Fng/8NmDt1VVATUX56mqrVU1U1XjXjY2Nk3c3FISVIzh12jQ5UZ3ZFIS1VIzH1cW5wJfqapnAqzSmLZ8nY8c/ZPfHdM+Ec
mOsw3Y1kcfJUnd/AW3JKmTYSFJ6mRYSJI6GRaSpE6GhSSpk2EhSepkWEiSOhkWkqROhoUkqZNhIU
nqZFhIkjoZFpKkToaFJKmTYSFJ6mRYSJI6GRaSpE6GhSSpU69hkWRZkluT/EmSR5P8P8VJLjk+xM8nj7X
t7aJsk1SSaTPJjk9KHtbGztH0+ycfYzSpL60PeVxdXAF6rqx4F3A48ClwK7qmotsKutA5wLrG2fzcC
1kr16wD4NMoJoPmOgJjJTAQsZ9hVvm/j3AA56dkfWuvzv3rwRh2AwB4o4tTRU2SrSt7OjBnM9VTjkXxr
AOB7YAZwVAxH7Aa7a3dEa8GZfoJ3FmNkZC29dDC5A3kQwemuz0pRP0Jzl5jVVcqpspmVBpeP1P1
MZV069XvQzYVVVrgV1V1tHeBcYG37ibWcKTAWsoTcSUbQfQzDqN93/HE9fcYwVq5Z6qr3UZV/c
uBzvBuqpS8kRtYFPDfk9Xyhe9r3YcDcvHdkHuazyoqnhamHmU9mFpS9zVZ/b4zEKamUm6/OSM8CkBrzqiVNKR4
j4XmnQnB8kgv4yWVmuM/QXGLMCDPC0/Hd2EnNEfqp56rVXapEeSHr2U/h2pWeOHcXx2
Y9b2E8ZGqXWUGdM+BzkjNJP2IXBe3dnZoVQmE9XHGrf3HPmcPuf1gmp6gSkekkX2R2v75jLDHOC9S2
3XbC5s8P5MOntgMmdzQ98w5fVkR4Ux7qqlSTDd3sV3BLw8wgoV0eGsG5VqzoVE8HAmbnurVAyK8kpFzrW
mCs9RXNE3wvUwxw3j0YbQqHZ1vWIsnUMG5IZc9Fu1bpHq9DFrzlmxJjBVbvLbLxWGbHUlbmbYGFoM8vDh
Acf4EDGK8MbJ9+Sz+CdXL2e7wYbfQswOD0A2jYEpvIFiTLhIh0hEP4d5Dwh9mNbXSrWRU1zp6KQf
JWgFLBM9UDE8i4dRt+wM6cVfsxLVxm6n7vbU+dBCqWTc0wP9u6HjF7umHgkhZw4TpKg8d/SLd
M/ZmCTMKWn2zPgSvOfoNCTU3KCG1eYkdK1nTZEvqwjPW5L3SwC/VQMxqNVJBW7Vxu+Hw8qFuZ7t
sN+yAOjQQKpd6f0xVdx5O20zAEcRYk+KU2aGhunXm/i5dk3vxHXhCCIxSv+Mz4Ssp+t5/ncSS/lSZn0hwCVqpb7tCvy
+sQlN7YVGIHE9PKTDSBTPn0PMMUqn3WFjf1BP3VcD2UOnNjc8fQ2HOgPOPoDYUyMvNvqTrhZsjmnpCJzZ
/KzLy5LH4Eb9qZeIqGHUhnvv4vhKDhtRDqkLOBtELScXSM/L4PRLz4JUsvOgzaV31gqAfYfy2i9r4nJJ
K2pjTPpbaJr6BNdxHaQgu6bWUoFl9p0B62t3pyp0kk7g3zfo7gvTHX7ZHEiAsbd2Wo26q1rpu7OfJUvmW1h
0nRDwfn7tEpHd7Ud8dfUAWGAZQ4Azbik79wHCzF0o2WSjKXQxrdpS2cQAb9vDVFUX
KGKXtSefOn8bd2o5ZDYO8mM7kuXwGjF3JbeAmX9HjBqjEG5Gm1n5MYfeXEAXXhJRmWxEqMjAQ
OmkW0Rq7MmCpv3RJAcIRdc9A9p+yo8e7xfMXqc+3ZiVd9zm8hAJZT7TB8BRAMpz8/bZKQb0whDm
Q72H5gQMI4CMtSWQg8z5JE3WnKJr3WDd/nN2IK9G+zcjvHvb2ZKOfvDMDiahzRh2tuwHjLpz5jq
9ngpcZ+S2XNE1hPOLVSemTmhpBLl4fEg3/1YGyF5MDaYb2ld2yhH8ewwb3Rq1yWV6BzJc2MfJZx
2PVGfuPKKtU4oOQgJ9+fVkBz5yj/NkHNfpYhmCMXF5gt83WYkzpxUcx+CPvDsNYO/UGP/cOP2x
HYPHWnMfe0UFBuUT1wOAukBMvvcSb1E5WBQAbTjaqKG2Vcgf7gKH6IRObhNfHLwnfjvb5wW3mbT
t5ZOJ7tKOyVPUdG82yHPAKtnOHYAXcn7Px8fFjYpSTV80WhIQDd/9bOzqgtq4E6cbdI4fy1k+RLFuE7
wjLS6s0IfwN7n/o5Fn4O6wzczZPNB9pvHwVhfX4ApVgJpX2JjhHwzY4BntMl9SMoH/JYeFWAdeI/dP
xx6vfAqG7prlSsdoHLpeQOvTdPrgORZRYpDm5fNf3f0kp/o3WXZJmrpCQxGtuAHR4TBKLrC7CRsR
xwl7Hv0f+p/QbNhf4mGowbAmv5UhQGrc1QSUW/eSxGNeoNqdj3CwP61JmRbn7lq61AExOZgPyY8
0nBUmB+yJ21pADWEbKf8GuJVHShuT6BhIfULL/gZkW+pFMg5KOwAxkUI8N+w7D+kBvBx/l4S0E5o0
ZnzsVFGm8Og/lDEcvbjN40i4jHTmx9YKqH1U9H3BKVHmuSIFzXrBdO2aE6e6kaxZzUpzhx9nXxTO7qPC
G7cd2a7/1V1b2Y4iKkw3FOXm6OsvyR7ZE3PQk3WsRRLewLzn6zS1t2ugObOv4YU9qSBU1CN3YgnoIB
oZ0IcRvOIkC3lQsXvnGk8uNHawwhdkPkzTcEqAMp4lXIQDlpcDHAP+wdK5CdaqzFVU1trjJbr7eNsZPI
AOoErXKC0SRAmfvjsXDOhe2u8Wb1Ri0QLNDYYlNugkzwuzgAmuvDUAT7IWL1KgNCnqpcq2kBbPt
TGDA3FpeLzuy2UmuWsN9dgy6t+9pF6RYd5vvTm1i4NJNkvSyvWbS+Qw33eq4d11wJxh1ZXeCCMHEBU
U6tXCUk5qqEoaPQPZG4yVDVlHjFQS05tXTSlM9tfV3lzFm7yPlrh3MstnDrKV+6aLLcS1hsFBmDNC
Tu4RJKCHm7N9ZsEYVjhyDdC1tjfN1zcUV6JAKxxD4+rg2+SLujJ5XV/qFQ/v/IP4YOHQ0U4x8MTXL

VWLsCwkKQlYNTHfYwDp7ZHckiSlphRf2fxEPAX+uyIJGnxGvXK4kTgkST3AC9PF6vq/b30SpK0qIwa
Fr/ZZyckSYvbqLOh/mffHZEkLV6jzoZ6kcHsJ4BjgTcA366qH+qrY5KkxWPUK4u3Ty9n8Oq7DcBZfXV
KkrS4fN9Pna2B/wycPUr7JEcnuT/JH7b1k5PcnWQyyc1Jjm31N7b1ybZ9zdAxPtrqjyUZ6bySpLkz6jDUzw
+tHsXgdxffGfEcvwo8yuDhgwAfB66qqpuS/AdgE3Bt+95fVe9MclFr97eTnApcBLyLwTOq/keSv1RVrxx8I
klSP0a9svi5oc/ZwIsMhqIOKckq4G8Cv9vWA7wPuLU12UQ6RBsk/Q7L1EFtlAFBMCRJRmwwt2LFdtD2RwZ46fU2bG+vhPUJ
+S5LmwKj3LC45zOP/W+CfAtP3PE4Anq+qA219N9971PlK4Kl2vgNJXmjtVwJ3DR1zeJ/vSrIZ2Azwjne8
4zC7K0mayagvP1qV5LYke9vn99tVw6H2+VvA3qq6b0562qGqNqhat3xD/HBdv9C3oxiHVxFPq3bXvGJFI0VdDT9q+jLkK3GskKdoIEaPFr1xe+a/Xcvfbh+H0QFoxqlxxg1pUV4WWFk9KRhYVcOv0qP6yO3+fgTZUZXR5NVqaNiXqbq
vhT9fUF6tMXWaEy8qtdC7eyIomMuoHIUe4T4MMu+2ck3+JqAPEZTQo1JKQ6C3Z5CMAN4lt6+P+aLVIzCjVujYA
ljfqUZ30rGdrQHZehYPptxcnAlhDyJmVU0M4e+rYKu3nCN5UHR5IVpGovhWVPr6XlwU0bFZfqgKnsq+AnOF
TV5Y/ma/Na7vdkOqLp5G1uYgd1O4ZI4+RapwP7xVpjWHDJW/cf2pvlE5uenKDp7jTkGq6jbDFq7Ei6nrlzj
ZlYjkDLPOkfbExtMqppV82CZVRPR0YGXvUGvRpUQEVJ4Kmo9yOgKnUFRZYGbW9k8W+UjtH0fE3lzYGbn
VWLsCwkKQlYNTHfYwDp7ZHckiSlphRf2fxEPAX+uyIJGnxGvXK4kTgkST3AC9PF6vq/b30SpK0qIwa
Fr/ZZyckSYvbqLOh/mffHZEkLV6jzoZ6kcHsJ4BjgTcA366qH+qrY5KkxWPUK4u3Ty9n8Oq7DcBZfXV
KkrS4fN9Pna2B/wycPUr7JEcnuT/JH7b1k5PcnWQyyc1Jjm31N7b1ybZ9zdAxPtrqjyUZ6bySpLkz6jDUzw
+tHsXgdxffGfEcvwo8yuDhgwAfB66qqpuS/AdgE3Bt+95fVe9MclFr97eTnApcBLyLwTOq/keSv1RVrxx8I
klSP0a9svi5oc/ZwIsMhqIOKckq4G8Cv9vWA7wPuLU12Q6c35Y3tHXa9nVDQ143VdXLVfU1YBI4Y8R
+S5LmwKj3LC45zOP/W+CfAtP3PE4Anq+qA219N9971PlK4Kl2vgNJXmjtVwJ3DR1zeJ/vSrIZ2Azwjne8
4zC7K0mayagvP1qV5LYke9vn99tVw6H2+VvA3qq6b0562qGqtlbVeFWNj42NzccpJWnJGHUY6npgB4N
7Bj8C/JdWO5T3Au9P8nXgJgbDT1cDy5JMGsAva05T3AaoC2/TjgueH6DPtIkubBqGExVlXXV9WB9
vk0cMj/fK+qj1bVqqpaw+AG9eer6u8u8CXwA+0JptBG5vyzvaOm3759uDC3cAF7XZUicDa4F7Ruy3JGkOj
BoWzyX5UJsGe3SSDzH4r/7D8RvAryeZPB4rpWvw44odV/HbgMoKoeBm4BHgE+B1zsqH+rHpw+hS9sAa8g
vqF4BPAlcx+CX3/wb+/qgnqaovAl9sy08ww2ymqvoOcMEs+18JXDnufkuSRDE5XGprbrdz3Af5YKvdB8SrDCOBXDGCj8qvlz7V
kGISJJe50Ydhvor00EBUFX7gPf00yVJ0mIzalgclWT59Eq7shj1qkSSdIQb9f/w/zXw5SSfZSXw5SSfbesX4D0ESVoyR
v0F9w1JJhj8VgLg56vqkqkK09BgWkqROhoUkqZNhIUnqZFhIkjoZFpKkTYSFJ6mRYSJI6GRaSpE6GhaSpk2EhSepkWEiSOhkWkqROhoUkqZNhIUnqZFhIkjoZFp
KkToaFJKmTYSFJ6mRYSJI6GRaSpE6GhSSpU29hkeRNSe5J8pJ8sdJHk7yW61+cpK7k0wmuTnJ7kvaxoz5mubPpvrk0gNLJlF7zNlx3M5X7NzNpWA3Ha3+rW0a2h1r6xX
bavGTrWR1v9sSRn99VnSdLM+ryyeBl4X1W9GzgdWAc4CC2JWcDHgauq6p6Z2Y
h4F3AO8KkkR/fYb0nSQXoLi85sq+zHz2/KqpVYx+ZwdT+TXC4ld+tHvT9kNR+cnu+8N4ZTsGYKpDIckq0wUkSzKyiNCUJRUGkb8/cPR72qgsbdSB+MKoRnk0gCsJZW/l
Z5FkqOTPADsBXYCfwo8X1UHWpPdWPdwMq2vBJ4H4Ckbp4G+z7D5D59qcZCLJxNTUVA9/jSQtXb2G
RVW9UlWnM6QASYXA38WI/n2lpV41U1PjY21tdpJGlJmpfZW9/X6X9X2pPAF4F4E6CtjTlvcAW1h3hG
uOA54brM+wjSZoHuHfc6GGkuyrC2/GfhZ4GE1oXGbj3VR/2c0IlCE/iqxjdptE3U9Hk8/mpe39lygnrkI
Z+rephOAnY3mYuHQXcUlV/mOQR4KYk/xF4ALiuNb8O+P0kk8B+BDfB2ha3j1XtR5gWKAkGnWD5tknCg1zeKj6X6H1G3UFZOw2A8lbdZsh1jqiLuyeX
Lkg7SW1hU1YPAe2aoP8EMs5mq6jvABBbMc60rgyrnuoyRpCNP6CW5LUybCQJHUyLCRJHUyLCQJHUyLCRJnQwLSVInw0
KS1MmwkCR1MiwkSZ0MC0lSJ8NCktTJsJAkdTIsJEmdDAtJUifDQpLUybCQJHUyLCRJnQwLSVInw0
KS1KnPd3CvTvKFJI8leTjJh1r9LUl2JNnVvpe3epJck2Syye5jjrmWBtH6fJpzlBaTpTrb6dYrEn
C7g0yanAZcZuqloL7GrAOcCa9tnM3AtDMIF2AKcyeB1MGO+HqVF6/fFJI8CK4ENwPmt2SYG8/74SKvf
0ZtuBLwK/0eo3VFUBdyZ1lSk1nZnVe0HSHIfBd1/vb3cx4ZZ9nXYLGe3M+fbxgbzMdS1AS6xXQUb3bljc8TzUN6yPvZ7KwUHHXUc
A9wN3AihYkAM8AK9ySuCpod12vK+MNtzgPRRF1qKSnuHK1VqjE5vSV6bPcN+2XtQwzUJY/pQwzTLU/wOsPk4+S1mJWTO4Pj4fj
8VbW1qsaranxsbwkDuilJanoNiyQ/xiAorqmqz7W23pzkwWqtPO9kIEyvOgRNDGSPEC6QaN2FhYq8iu+zYndL4ZcNXkEXXNNV2nDz6zrOgzbyH9+M9
qv7N0KYdwPSMpo3A7UP1i9usqLOAF9pw1Z3A+iTL243v9a0mSZ1EQ20cCABDMW6AonE4oWgFCDFP
XJJuBJ4MK27Q7gPGASeAm4BKCq9iW5Ari3tbt9+mZ3X/gR/sC1FJOTYa3kg9Q+4xDbpgfScq4XDpvZLRmWA9onKl5vA5HgDmMv
8Pf8EtSepkWEiSOhkWkqROhoUkqZNhIUnqZFhIkjoZFpKkToaFJKmTYSFJ6mRYSJI6GRaSpE6GhSSp
k2EhSepkWEiSOhkWkqROhoUkqZNhIUnq1Oc7uLcl2ZvkvaoHa8Ul2JnvkoaHa8Ul2Jnm8fS9v9SS5JslkkgeTnD60z8bW/v
EkG2c6lySpX31eWXwaOOeg2mXArqpaC+xqwDnAmcAWyZDhhJ0vzpLSyq6kvA/hnlDcCmtrwJOG+ofkMN3AksS3IScA6wo6r2V9VzwA5eH0CSpJ7N9zu
kvAvoPKG4DtXk7Xtq6r9bW23tQwlv6r9wE5ePSVE5J7NN9z2LFVX1dFt+BljRllcCTw212932m63+Okk2J5lIMrFv375577gkLWaH7yfmpJzOoucMaQD3FYKv2129
1qs9VfI8nmJBNqpampua21vAJ72BkwDssCP/tEnMf6kjyIRUOoDf0G7v4pvVNQfe3c0Y0dD5eEW3XswFK35ZqrCmKhJ5tvd
uVavNVpckzaP5DosdwMa2vBG4d6i+qc2KOgt4oQ1X3QasS7K83dhe12qSpHnU27zfJJ8GzgPOTbKbwaymq4Gbk1wPPAlcAH0
NZTR8DbkmyXgWuLAk0oZpYLWzrBmMvXBE9K99mJ2bTueCVGXELIA2Z5t7kCmSEdv7lvEVwKfF9tdQ0m0xS7GptSj2vfSqyK+jr9Ja/l2J
xtgHb5rBrkqTvk7/gliR1MiwkSZ0MC0lSJ0MC0l

IvZp4JyF7sRSYljoUHzMihalqvoSsG+h+7GUGBY6lJkes7JygfoiaQEZFpKkToaFDsXHrEgCDAsdmo9Zk
QQYFjqEqjoATD9m5VHgFh+zosUgyY3Al4FTkuxOsmmh+/R65+M+JEmdvLKQJHUyLCRJnQwLSVIn
w0KS1MmwkCR1Miykw5DklSQPDH0O+UTeJHckWdY+v3QY5/vNJP/k8Hss/WCOiNeqSovQ/62q00Zt
XFXnASRZA/wS8Kl+uiX1wysLaY4kOa69++OUtn5jkn/Ylr+e5ETgY8CPtquR32nbPpLk3iQPJvmtoeP98y
T/J8kfAacswJ8kfZdXFtLheXOSB4bW/1VV3Zzkl4FPJ7kaWF5V//Gg/S4DfmL6qiTJemAtg8fBB9iR5KeB
bzN4vMppDP53+hXgvv7+HOnQDAvp8Mw4DFVVO5NcwOClUe8e4Tjr2+f+tv42BuHxduC2qnoJIInP5N
KCchhKmkNJjgJ+HHgJWD7KLgyuSk5rn3dW1XW9dlI6DIaFNLd+jcFDF/8OcH2SNxy0/UUGVw3T7gR
+IcnbAJKsTPLDwJeA85O8OcnbgZ/rv+vS7ByGkg7PwfcsPgdcD/wD4IyqejHJl4B/AWyqblRVzyX5X0ke
Av5lO/A+2Z0j36VccUAAAAAElFTkSuQmCC\n",
      "text/plain": [
       "<Figure size 432x288 with 1 Axes>"
      ]
     },
     "metadata": {
      "needs_background": "light"
     },
     "output_type": "display_data"
    }
   ],
   "source": [
    "sns.countplot(df['Exited'])"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 18,
   "id": "c77c724c",
   "metadata": {
    "execution": {
     "iopub.execute_input": "2021-12-21T16:51:28.715665Z",
     "iopub.status.busy": "2021-12-21T16:51:28.714984Z",
     "iopub.status.idle": "2021-12-21T16:51:28.718504Z",
     "shell.execute_reply": "2021-12-21T16:51:28.717824Z",
     "shell.execute_reply.started": "2021-12-21T16:22:23.241922Z"
    },
    "papermill": {
     "duration": 0.044497,
     "end_time": "2021-12-21T16:51:28.718649",
     "exception": false,
     "start_time": "2021-12-21T16:51:28.674152",
     "status": "completed"

```
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "0    7963\n",
     "1    2037\n",
     "Name: Exited, dtype: int64"
    ]
   },
   "execution_count": 18,
   "metadata": {},
   "output_type": "execute_result"
  }
 ],
 "source": [
  "df['Exited'].value_counts()"
 ]
},
{
 "cell_type": "code",
 "execution_count": 19,
 "id": "f4ba6a1f",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:28.793562Z",
   "iopub.status.busy": "2021-12-21T16:51:28.792872Z",
   "iopub.status.idle": "2021-12-21T16:51:28.798414Z",
   "shell.execute_reply": "2021-12-21T16:51:28.797875Z",
   "shell.execute_reply.started": "2021-12-21T16:22:23.251339Z"
  },
  "papermill": {
   "duration": 0.045252,
   "end_time": "2021-12-21T16:51:28.798555",
   "exception": false,
   "start_time": "2021-12-21T16:51:28.753303",
   "status": "completed"
  },
  "tags": []
 },
```

```
 "outputs": [],
 "source": [
  "#splitting data to input and output\n",
  "X=df.drop('Exited',axis=1) #input\n",
  "y=df['Exited']  #output(label)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 20,
 "id": "2257d50b",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:28.875644Z",
   "iopub.status.busy": "2021-12-21T16:51:28.874865Z",
   "iopub.status.idle": "2021-12-21T16:51:28.880854Z",
   "shell.execute_reply": "2021-12-21T16:51:28.880316Z",
   "shell.execute_reply.started": "2021-12-21T16:22:23.362068Z"
  },
  "papermill": {
   "duration": 0.047357,
   "end_time": "2021-12-21T16:51:28.881009",
   "exception": false,
   "start_time": "2021-12-21T16:51:28.833652",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "X_train, X_test, y_train, y_test= train_test_split(X,y,test_size=0.2,shuffle=True)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 21,
 "id": "0799924a",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:28.955883Z",
   "iopub.status.busy": "2021-12-21T16:51:28.955093Z",
   "iopub.status.idle": "2021-12-21T16:51:28.958213Z",
```

```
   "shell.execute_reply": "2021-12-21T16:51:28.958731Z",
   "shell.execute_reply.started": "2021-12-21T16:22:23.589379Z"
  },
  "papermill": {
   "duration": 0.04341,
   "end_time": "2021-12-21T16:51:28.958900",
   "exception": false,
   "start_time": "2021-12-21T16:51:28.915490",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "(10000, 13)\n",
    "(10000,)\n"
   ]
  }
 ],
 "source": [
  "print(X.shape)\n",
  "print(y.shape)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 22,
 "id": "a7e5734c",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:29.034485Z",
   "iopub.status.busy": "2021-12-21T16:51:29.033527Z",
   "iopub.status.idle": "2021-12-21T16:51:29.038550Z",
   "shell.execute_reply": "2021-12-21T16:51:29.037927Z",
   "shell.execute_reply.started": "2021-12-21T16:22:23.955086Z"
  },
  "papermill": {
   "duration": 0.045016,
   "end_time": "2021-12-21T16:51:29.038692",
```

```
   "exception": false,
   "start_time": "2021-12-21T16:51:28.993676",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    " X_train.shape :  (8000, 13)\n",
    " y_train.shape :  (8000,)\n",
    " X_test.shape :  (2000, 13)\n",
    " y_test.shape :  (2000,)\n"
   ]
  }
 ],
 "source": [
  "print(' X_train.shape : ',X_train.shape)\n",
  "print(' y_train.shape : ',y_train.shape)\n",
  "print(' X_test.shape : ',X_test.shape)\n",
  "print(' y_test.shape : ',y_test.shape)"
 ]
},
{
 "cell_type": "markdown",
 "id": "d53cfe60",
 "metadata": {
  "papermill": {
   "duration": 0.034608,
   "end_time": "2021-12-21T16:51:29.108819",
   "exception": false,
   "start_time": "2021-12-21T16:51:29.074211",
   "status": "completed"
  },
  "tags": []
 },
 "source": [
  "# deep learning ANN"
 ]
},
```

```
{
 "cell_type": "code",
 "execution_count": 23,
 "id": "f2ddcb87",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:51:29.186141Z",
   "iopub.status.busy": "2021-12-21T16:51:29.185171Z",
   "iopub.status.idle": "2021-12-21T16:51:29.317013Z",
   "shell.execute_reply": "2021-12-21T16:51:29.317542Z",
   "shell.execute_reply.started": "2021-12-21T16:22:24.415566Z"
  },
  "papermill": {
   "duration": 0.174048,
   "end_time": "2021-12-21T16:51:29.317717",
   "exception": false,
   "start_time": "2021-12-21T16:51:29.143669",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "Model: \"sequential\"\n",
    "_________________________________________________________________\n",
    "Layer (type)                 Output Shape              Param #   \n",
    "=================================================================\n",
    "dense (Dense)                (None, 6)                 84        \n",
    "_________________________________________________________________\n",
    "dense_1 (Dense)              (None, 5)                 35        \n",
    "_________________________________________________________________\n",
    "dense_2 (Dense)              (None, 1)                 6         \n",
    "=================================================================\n",
    "Total params: 125\n",
    "Trainable params: 125\n",
    "Non-trainable params: 0\n",
    "_________________________________________________________________\n"
   ]
  },
```

```
{
 "name": "stderr",
 "output_type": "stream",
 "text": [
  "\n",
  "User settings:\n",
  "\n",
  "   KMP_AFFINITY=granularity=fine,verbose,compact,1,0\n",
  "   KMP_BLOCKTIME=0\n",
  "   KMP_DUPLICATE_LIB_OK=True\n",
  "   KMP_INIT_AT_FORK=FALSE\n",
  "   KMP_SETTINGS=1\n",
  "   KMP_WARNINGS=0\n",
  "\n",
  "Effective settings:\n",
  "\n",
  "   KMP_ABORT_DELAY=0\n",
  "   KMP_ADAPTIVE_LOCK_PROPS='1,1024'\n",
  "   KMP_ALIGN_ALLOC=64\n",
  "   KMP_ALL_THREADPRIVATE=128\n",
  "   KMP_ATOMIC_MODE=2\n",
  "   KMP_BLOCKTIME=0\n",
  "   KMP_CPUINFO_FILE: value is not defined\n",
  "   KMP_DETERMINISTIC_REDUCTION=false\n",
  "   KMP_DEVICE_THREAD_LIMIT=2147483647\n",
  "   KMP_DISP_NUM_BUFFERS=7\n",
  "   KMP_DUPLICATE_LIB_OK=true\n",
  "   KMP_ENABLE_TASK_THROTTLING=true\n",
  "   KMP_FORCE_REDUCTION: value is not defined\n",
  "   KMP_FOREIGN_THREADS_THREADPRIVATE=true\n",
  "   KMP_FORKJOIN_BARRIER='2,2'\n",
  "   KMP_FORKJOIN_BARRIER_PATTERN='hyper,hyper'\n",
  "   KMP_GTID_MODE=3\n",
  "   KMP_HANDLE_SIGNALS=false\n",
  "   KMP_HOT_TEAMS_MAX_LEVEL=1\n",
  "   KMP_HOT_TEAMS_MODE=0\n",
  "   KMP_INIT_AT_FORK=true\n",
  "   KMP_LIBRARY=throughput\n",
  "   KMP_LOCK_KIND=queuing\n",
  "   KMP_MALLOC_POOL_INCR=1M\n",
  "   KMP_NUM_LOCKS_IN_BLOCK=1\n",
  "   KMP_PLAIN_BARRIER='2,2'\n",
```

```
"   KMP_PLAIN_BARRIER_PATTERN='hyper,hyper'\n",
"   KMP_REDUCTION_BARRIER='1,1'\n",
"   KMP_REDUCTION_BARRIER_PATTERN='hyper,hyper'\n",
"   KMP_SCHEDULE='static,balanced;guided,iterative'\n",
"   KMP_SETTINGS=true\n",
"   KMP_SPIN_BACKOFF_PARAMS='4096,100'\n",
"   KMP_STACKOFFSET=64\n",
"   KMP_STACKPAD=0\n",
"   KMP_STACKSIZE=8M\n",
"   KMP_STORAGE_MAP=false\n",
"   KMP_TASKING=2\n",
"   KMP_TASKLOOP_MIN_TASKS=0\n",
"   KMP_TASK_STEALING_CONSTRAINT=1\n",
"   KMP_TEAMS_THREAD_LIMIT=4\n",
"   KMP_TOPOLOGY_METHOD=all\n",
"   KMP_USE_YIELD=1\n",
"   KMP_VERSION=false\n",
"   KMP_WARNINGS=false\n",
"   OMP_AFFINITY_FORMAT='OMP: pid %P tid %i thread %n bound to OS proc set {%A}'\n",
"   OMP_ALLOCATOR=omp_default_mem_alloc\n",
"   OMP_CANCELLATION=false\n",
"   OMP_DEFAULT_DEVICE=0\n",
"   OMP_DISPLAY_AFFINITY=false\n",
"   OMP_DISPLAY_ENV=false\n",
"   OMP_DYNAMIC=false\n",
"   OMP_MAX_ACTIVE_LEVELS=1\n",
"   OMP_MAX_TASK_PRIORITY=0\n",
"   OMP_NESTED: deprecated; max-active-levels-var=1\n",
"   OMP_NUM_THREADS: value is not defined\n",
"   OMP_PLACES: value is not defined\n",
"   OMP_PROC_BIND='intel'\n",
"   OMP_SCHEDULE='static'\n",
"   OMP_STACKSIZE=8M\n",
"   OMP_TARGET_OFFLOAD=DEFAULT\n",
"   OMP_THREAD_LIMIT=2147483647\n",
"   OMP_WAIT_POLICY=PASSIVE\n",
"   KMP_AFFINITY='verbose,warnings,respect,granularity=fine,compact,1,0'\n",
  "\n",
   "2021-12-21 16:51:29.218493: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.\n"
 ]
```

```
   }
  ],
  "source": [
   "model = Sequential()\n",
   "model.add(Dense(6, input_dim=13, activation='relu'))\n",
   "\n",
   "model.add(Dense(5, activation='relu'))\n",
   "\n",
   "model.add(Dense(1, activation='sigmoid'))\n",
   "\n",
   "model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])\n",
   "model.summary()"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 24,
  "id": "7bfd3f2e",
  "metadata": {
   "execution": {
    "iopub.execute_input": "2021-12-21T16:51:29.396105Z",
    "iopub.status.busy": "2021-12-21T16:51:29.395447Z",
    "iopub.status.idle": "2021-12-21T16:53:52.009314Z",
    "shell.execute_reply": "2021-12-21T16:53:52.008618Z",
    "shell.execute_reply.started": "2021-12-21T16:22:24.612490Z"
   },
   "papermill": {
    "duration": 142.65612,
    "end_time": "2021-12-21T16:53:52.009498",
    "exception": false,
    "start_time": "2021-12-21T16:51:29.353378",
    "status": "completed"
   },
   "tags": []
  },
  "outputs": [
   {
    "name": "stderr",
    "output_type": "stream",
    "text": [
     "2021-12-21 16:51:29.482130: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optimization Passes are enabled (registered 2)\n"
```

```
   ]
  },
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "Epoch 1/100\n",
     "680/680 [==============================] - 2s 2ms/step - loss: 0.4925 - accuracy: 0.7728 - val_loss: 0.4564 - val_accuracy: 0.7825\n",
    "Epoch 2/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.4351 - accuracy: 0.8031 - val_loss: 0.4400 - val_accuracy: 0.8033\n",
    "Epoch 3/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.4233 - accuracy: 0.8143 - val_loss: 0.4335 - val_accuracy: 0.8150\n",
    "Epoch 4/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.4164 - accuracy: 0.8216 - val_loss: 0.4302 - val_accuracy: 0.8233\n",
    "Epoch 5/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.4121 - accuracy: 0.8254 - val_loss: 0.4280 - val_accuracy: 0.8275\n",
    "Epoch 6/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.4086 - accuracy: 0.8297 - val_loss: 0.4246 - val_accuracy: 0.8283\n",
    "Epoch 7/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.4057 - accuracy: 0.8331 - val_loss: 0.4229 - val_accuracy: 0.8292\n",
    "Epoch 8/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.4028 - accuracy: 0.8343 - val_loss: 0.4192 - val_accuracy: 0.8275\n",
    "Epoch 9/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.3997 - accuracy: 0.8338 - val_loss: 0.4177 - val_accuracy: 0.8317\n",
    "Epoch 10/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.3960 - accuracy: 0.8338 - val_loss: 0.4149 - val_accuracy: 0.8333\n",
    "Epoch 11/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.3930 - accuracy: 0.8372 - val_loss: 0.4096 - val_accuracy: 0.8383\n",
    "Epoch 12/100\n",
     "680/680 [==============================] - 1s 2ms/step - loss: 0.3877 - accuracy: 0.8379 - val_loss: 0.4071 - val_accuracy: 0.8367\n",
```

"Epoch 13/100\n",
"680/680 [==============================] - 2s 2ms/step - loss: 0.3827 - accuracy: 0.8397 - val_loss: 0.3988 - val_accuracy: 0.8383\n",
"Epoch 14/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3781 - accuracy: 0.8413 - val_loss: 0.3974 - val_accuracy: 0.8342\n",
"Epoch 15/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3745 - accuracy: 0.8441 - val_loss: 0.3948 - val_accuracy: 0.8367\n",
"Epoch 16/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3719 - accuracy: 0.8447 - val_loss: 0.3936 - val_accuracy: 0.8358\n",
"Epoch 17/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3703 - accuracy: 0.8421 - val_loss: 0.3941 - val_accuracy: 0.8383\n",
"Epoch 18/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3680 - accuracy: 0.8431 - val_loss: 0.3911 - val_accuracy: 0.8375\n",
"Epoch 19/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3664 - accuracy: 0.8453 - val_loss: 0.3910 - val_accuracy: 0.8342\n",
"Epoch 20/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3653 - accuracy: 0.8437 - val_loss: 0.3918 - val_accuracy: 0.8325\n",
"Epoch 21/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3641 - accuracy: 0.8443 - val_loss: 0.3927 - val_accuracy: 0.8383\n",
"Epoch 22/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3632 - accuracy: 0.8449 - val_loss: 0.3913 - val_accuracy: 0.8400\n",
"Epoch 23/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3616 - accuracy: 0.8456 - val_loss: 0.3917 - val_accuracy: 0.8400\n",
"Epoch 24/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3615 - accuracy: 0.8440 - val_loss: 0.3899 - val_accuracy: 0.8375\n",
"Epoch 25/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3602 - accuracy: 0.8462 - val_loss: 0.3915 - val_accuracy: 0.8367\n",
"Epoch 26/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3599 - accuracy: 0.8468 - val_loss: 0.3892 - val_accuracy: 0.8342\n",

"Epoch 27/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3590 - accuracy: 0.8456 - val_loss: 0.3913 - val_accuracy: 0.8400\n",
"Epoch 28/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3585 - accuracy: 0.8484 - val_loss: 0.3876 - val_accuracy: 0.8342\n",
"Epoch 29/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3585 - accuracy: 0.8460 - val_loss: 0.3872 - val_accuracy: 0.8350\n",
"Epoch 30/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3576 - accuracy: 0.8456 - val_loss: 0.3881 - val_accuracy: 0.8383\n",
"Epoch 31/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3578 - accuracy: 0.8463 - val_loss: 0.3885 - val_accuracy: 0.8383\n",
"Epoch 32/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3577 - accuracy: 0.8479 - val_loss: 0.3931 - val_accuracy: 0.8342\n",
"Epoch 33/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3570 - accuracy: 0.8476 - val_loss: 0.3902 - val_accuracy: 0.8367\n",
"Epoch 34/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3569 - accuracy: 0.8463 - val_loss: 0.3898 - val_accuracy: 0.8342\n",
"Epoch 35/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3561 - accuracy: 0.8485 - val_loss: 0.3882 - val_accuracy: 0.8375\n",
"Epoch 36/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3563 - accuracy: 0.8479 - val_loss: 0.3863 - val_accuracy: 0.8375\n",
"Epoch 37/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3562 - accuracy: 0.8478 - val_loss: 0.3875 - val_accuracy: 0.8417\n",
"Epoch 38/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3561 - accuracy: 0.8469 - val_loss: 0.3873 - val_accuracy: 0.8375\n",
"Epoch 39/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3552 - accuracy: 0.8500 - val_loss: 0.3916 - val_accuracy: 0.8342\n",
"Epoch 40/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3555 - accuracy: 0.8493 - val_loss: 0.3871 - val_accuracy: 0.8367\n",

"Epoch 41/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3544 - accuracy: 0.8482 - val_loss: 0.3879 - val_accuracy: 0.8350\n",
"Epoch 42/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3549 - accuracy: 0.8491 - val_loss: 0.3858 - val_accuracy: 0.8367\n",
"Epoch 43/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3545 - accuracy: 0.8491 - val_loss: 0.3898 - val_accuracy: 0.8342\n",
"Epoch 44/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3542 - accuracy: 0.8471 - val_loss: 0.3873 - val_accuracy: 0.8358\n",
"Epoch 45/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3537 - accuracy: 0.8497 - val_loss: 0.3891 - val_accuracy: 0.8383\n",
"Epoch 46/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3532 - accuracy: 0.8490 - val_loss: 0.3881 - val_accuracy: 0.8375\n",
"Epoch 47/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3530 - accuracy: 0.8479 - val_loss: 0.3869 - val_accuracy: 0.8350\n",
"Epoch 48/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3522 - accuracy: 0.8506 - val_loss: 0.3870 - val_accuracy: 0.8350\n",
"Epoch 49/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3523 - accuracy: 0.8506 - val_loss: 0.3849 - val_accuracy: 0.8375\n",
"Epoch 50/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3517 - accuracy: 0.8507 - val_loss: 0.3862 - val_accuracy: 0.8358\n",
"Epoch 51/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3513 - accuracy: 0.8525 - val_loss: 0.3857 - val_accuracy: 0.8375\n",
"Epoch 52/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3509 - accuracy: 0.8519 - val_loss: 0.3884 - val_accuracy: 0.8308\n",
"Epoch 53/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3509 - accuracy: 0.8531 - val_loss: 0.3849 - val_accuracy: 0.8367\n",
"Epoch 54/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3501 - accuracy: 0.8538 - val_loss: 0.3838 - val_accuracy: 0.8367\n",

"Epoch 55/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3494 - accuracy: 0.8534 - val_loss: 0.3846 - val_accuracy: 0.8392\n",
"Epoch 56/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3487 - accuracy: 0.8528 - val_loss: 0.3804 - val_accuracy: 0.8392\n",
"Epoch 57/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3482 - accuracy: 0.8522 - val_loss: 0.3789 - val_accuracy: 0.8392\n",
"Epoch 58/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3475 - accuracy: 0.8556 - val_loss: 0.3781 - val_accuracy: 0.8433\n",
"Epoch 59/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3484 - accuracy: 0.8531 - val_loss: 0.3784 - val_accuracy: 0.8433\n",
"Epoch 60/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3467 - accuracy: 0.8557 - val_loss: 0.3827 - val_accuracy: 0.8417\n",
"Epoch 61/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3472 - accuracy: 0.8574 - val_loss: 0.3764 - val_accuracy: 0.8433\n",
"Epoch 62/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3454 - accuracy: 0.8546 - val_loss: 0.3772 - val_accuracy: 0.8417\n",
"Epoch 63/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3455 - accuracy: 0.8549 - val_loss: 0.3791 - val_accuracy: 0.8433\n",
"Epoch 64/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3442 - accuracy: 0.8554 - val_loss: 0.3735 - val_accuracy: 0.8442\n",
"Epoch 65/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3435 - accuracy: 0.8568 - val_loss: 0.3709 - val_accuracy: 0.8483\n",
"Epoch 66/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3437 - accuracy: 0.8578 - val_loss: 0.3715 - val_accuracy: 0.8508\n",
"Epoch 67/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3427 - accuracy: 0.8547 - val_loss: 0.3702 - val_accuracy: 0.8483\n",
"Epoch 68/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3423 - accuracy: 0.8571 - val_loss: 0.3702 - val_accuracy: 0.8483\n",

"Epoch 69/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3418 - accuracy: 0.8556 - val_loss: 0.3688 - val_accuracy: 0.8467\n",
"Epoch 70/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3409 - accuracy: 0.8576 - val_loss: 0.3700 - val_accuracy: 0.8458\n",
"Epoch 71/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3416 - accuracy: 0.8571 - val_loss: 0.3732 - val_accuracy: 0.8517\n",
"Epoch 72/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3417 - accuracy: 0.8574 - val_loss: 0.3686 - val_accuracy: 0.8500\n",
"Epoch 73/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3406 - accuracy: 0.8581 - val_loss: 0.3681 - val_accuracy: 0.8525\n",
"Epoch 74/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3401 - accuracy: 0.8576 - val_loss: 0.3646 - val_accuracy: 0.8500\n",
"Epoch 75/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3391 - accuracy: 0.8565 - val_loss: 0.3672 - val_accuracy: 0.8533\n",
"Epoch 76/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3403 - accuracy: 0.8579 - val_loss: 0.3672 - val_accuracy: 0.8483\n",
"Epoch 77/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3397 - accuracy: 0.8563 - val_loss: 0.3698 - val_accuracy: 0.8533\n",
"Epoch 78/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3388 - accuracy: 0.8600 - val_loss: 0.3712 - val_accuracy: 0.8525\n",
"Epoch 79/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3383 - accuracy: 0.8579 - val_loss: 0.3648 - val_accuracy: 0.8558\n",
"Epoch 80/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3367 - accuracy: 0.8588 - val_loss: 0.3693 - val_accuracy: 0.8542\n",
"Epoch 81/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3376 - accuracy: 0.8600 - val_loss: 0.3637 - val_accuracy: 0.8575\n",
"Epoch 82/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3374 - accuracy: 0.8601 - val_loss: 0.3639 - val_accuracy: 0.8558\n",

"Epoch 83/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3365 - accuracy: 0.8610 - val_loss: 0.3641 - val_accuracy: 0.8558\n",
"Epoch 84/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3359 - accuracy: 0.8618 - val_loss: 0.3616 - val_accuracy: 0.8558\n",
"Epoch 85/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3363 - accuracy: 0.8609 - val_loss: 0.3612 - val_accuracy: 0.8517\n",
"Epoch 86/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3353 - accuracy: 0.8600 - val_loss: 0.3604 - val_accuracy: 0.8517\n",
"Epoch 87/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3354 - accuracy: 0.8588 - val_loss: 0.3578 - val_accuracy: 0.8533\n",
"Epoch 88/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3355 - accuracy: 0.8606 - val_loss: 0.3597 - val_accuracy: 0.8533\n",
"Epoch 89/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3353 - accuracy: 0.8591 - val_loss: 0.3596 - val_accuracy: 0.8558\n",
"Epoch 90/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3352 - accuracy: 0.8588 - val_loss: 0.3594 - val_accuracy: 0.8583\n",
"Epoch 91/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3346 - accuracy: 0.8569 - val_loss: 0.3605 - val_accuracy: 0.8525\n",
"Epoch 92/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3344 - accuracy: 0.8601 - val_loss: 0.3635 - val_accuracy: 0.8550\n",
"Epoch 93/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3344 - accuracy: 0.8622 - val_loss: 0.3595 - val_accuracy: 0.8475\n",
"Epoch 94/100\n",
"680/680 [==============================] - 2s 2ms/step - loss: 0.3342 - accuracy: 0.8597 - val_loss: 0.3597 - val_accuracy: 0.8583\n",
"Epoch 95/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3338 - accuracy: 0.8597 - val_loss: 0.3609 - val_accuracy: 0.8533\n",
"Epoch 96/100\n",
"680/680 [==============================] - 1s 2ms/step - loss: 0.3341 - accuracy: 0.8587 - val_loss: 0.3578 - val_accuracy: 0.8533\n",

```
      "Epoch 97/100\n",
      "680/680 [==============================] - 1s 2ms/step - loss: 0.3340 - accuracy: 0.8612 - val_loss: 0.3600 - val_accuracy: 0.8558\n",
      "Epoch 98/100\n",
      "680/680 [==============================] - 1s 2ms/step - loss: 0.3335 - accuracy: 0.8596 - val_loss: 0.3574 - val_accuracy: 0.8542\n",
      "Epoch 99/100\n",
      "680/680 [==============================] - 1s 2ms/step - loss: 0.3333 - accuracy: 0.8609 - val_loss: 0.3586 - val_accuracy: 0.8500\n",
      "Epoch 100/100\n",
      "680/680 [==============================] - 1s 2ms/step - loss: 0.3334 - accuracy: 0.8596 - val_loss: 0.3615 - val_accuracy: 0.8558\n"
     ]
    }
   ],
   "source": [
    "history=model.fit(X_train, y_train, batch_size = 10, epochs = 100,validation_split=0.15)"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 25,
   "id": "9a7e12ed",
   "metadata": {
    "execution": {
     "iopub.execute_input": "2021-12-21T16:53:53.563792Z",
     "iopub.status.busy": "2021-12-21T16:53:53.547481Z",
     "iopub.status.idle": "2021-12-21T16:53:53.878163Z",
     "shell.execute_reply": "2021-12-21T16:53:53.877637Z",
     "shell.execute_reply.started": "2021-12-21T16:24:32.156781Z"
    },
    "papermill": {
     "duration": 1.080701,
     "end_time": "2021-12-21T16:53:53.878324",
     "exception": false,
     "start_time": "2021-12-21T16:53:52.797623",
     "status": "completed"
    },
    "tags": []
   },
   "outputs": [
    {
```

"data": {
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAAskAAAF1CAYAAAAa1Xd+AAAAOXRFWHRTb2Z0d2FyZQB
NYXRwbG90bGliIHZlcnNpb24zLjUuMCwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy8/fFQqAAAACXB
IWXMAAAsTAAALEwEAmpwYAACkM0lEQVR4nOzdd3jUVdbA8e+dSe8JKRAChA6BUENHmg0Fx
YaKoqKubW3r6qruq6vva1ty64NO4uCgCIIgtJr6IQWAiEkJCGkz9z3jztDJiGdJJNMzud58mTmV88
MOhzunHuu0lojhBBCCCGEKGNxdwBCCCGEEEI0N5IkCyGEEEIIUYEkyUIIIYQQQlQgSbIQQghhB
AVSJIshBBCCCFEBZIkCyGEEEIIUYEkyS2QUp7pdQNDX2sOyml9imlzmmE6y5RSv3O8fhapdTC2hx
bj/t0VErlKaWs9Y1VCCEqks/7Ol1XPu9Fg5IkuYk4/ody/tiVVUgUzz6vtiVqpDdwD6itg7TWoAjnf
ZHqmUKlZK9a3ttbTWH2utzh4Uip2a4vhCi4cjnff3I5z0opbSRSqlv
LP9NClE38nlfb/J5L5oVSZLdTCk1TimTxP1SQ1joIOABc5LLtY+dxLPbeyPOk1knGRdgjmtCfApha
q95Dh2r1Lqgnoe21kptVQdUIp9aNS6g2l1EdVxF2bbGGJ9USOUqp1wHlXXF2bGJ9USOUqp1wHlsq+rUuonR3zHlFIfK6XCHPs+BDoC3zpGhh5U
SsU7Rg4HMfKqXmKKWylTtz7caXU50/qp/zrem2SlVKKf7lVK3NVoVq8lKFfzbUMZ7GrFJHVN/1zQ4SPwWyUZq9uyUVxAs8/NFkXWY6SBc8D
lCewvFC7UDYeEOkJP9p1SqtOzRl7l9NIoJqtlXyoHSa1+jD/yy7OGmHDqqDfqVrRi7x6DdZLFKFHeTVrJcWalOSgQAWo1ZQ/WXcgkTJ
HQ9+fGVwxA5LQKUsHycoZcLKPN0pt80XKL1gknlgxJhSiuFTo64aHsG2AmEC1cf8EZb
mFtL4Rqr/WEVNwYTgXvyZ1FFP3TlNrPdPRNhnoD6YsXzhmmGYDv9gKCWVXJ2lG86xvnY1mKJM
ohzunHuu0lojhBBCCCGEKGNxdwBCCCCGEEEI0N5IkCyGEEEIIUYEkyUIIIYQQQlQgSbIQQghhB
AVSJIshBBCCCFEBZIkCyGEEEIIUYEkyS2QUp7pdQNDX2sOyml9imlzmmE6y5RSv3O8fhapdTC2hx
bj/t0VErlKaWs9Y1VCCEqks/7Ol1XPu9Fg5IkuYk4/ody/tiVUgUuz6+ty7W01hdorf/T0Mc2R0qpyv
ZHqmUKlZK9a3ttbTWH2utzh4Uip2a4vhCi4cjnff3I5z0opbSRSqlt
DX1fUjyTJTcTxP1SQ1joIOABc5LLtY+dxSikv90XZLH0EjFRKda6w/Wpgs9Z6ixQCv
LP9NClE38nlfb/J5L5oVSZLdTCk1TimVqpR6SCl1BPi3UipcKfWdUuLCHE3XPu4yR31XsUSHCjPJu0Kd
mVh
LsdFK6XylVJRNbxeITyOfN7L530tP+8zhe6wZOzsxcDXxZZOeMPT6mVffC71I3KxTm4WkK9x1U7BvMYAEd0MczL5Lq0sjpbKzxNo1e57lsXBcqWFW1Yls2sxjcJWmvKSRvWCAY6KnQTA40zphEJn9u6r5kyF1ZmkGVThj0NYVtlLhpNMKhnX84HCPIAZGNH9ya4d9+r3fW+uVhmDNM80rNhw7NbgTlq4IDOeXN6hYp6GBHcB1mIc9yMVtpLC5xv3IOq4e1S09m+7e/a0utxt3FUTwYT8gDz8WDO7OY3PSC01IAdlj5r4pCa3Oig2WBb1s1Yqj9+vaT
jFKgjb4hyVdKjQRvuYAZ3NcSTeYVHKvPg6V1yiphEkG9DtD+8fJDqGO0xdx4hdhh4Mn8m8qMhcNdL4K7UD8mEYT4ru2qrS8eppXKTOzf0OwHh+QHzYn1WGqIoZf36vvlz4zQa3PwGb3Ia74wF/lXL6nQhPM2t8YaQQfUFOMgo4OzPPT/dg4TTKxFHRwyGNOSlR53/xkPbsC3Bbpxf13VpVMEyy/g75+3n55Zn+fb2IUxQ+zM97TxErZfrhm2BgfE4WbeFl5kvMlY9m7fH5+ZXOx1qKsG/Qq6aRnn5fKVz9nGZaX2P8SDZqHD7cjA6r66JlfFG3kWWjhP1oI5SOc2bhSsNm5Jm4ovHMrgFbbdBZxb/+ueVTJbyy+ecF6pE9wEoWb0vqdlVh5O3SzPY8S2DZ7xIFQ+4Vwvf/WdfyyxV6ZQ3VN2OK9TSwgXPmI8xxPu1ofqJmMjgEu58EN1q7p6xPmjpRvDaK7VE0gS/93gyFAjK+xNy5j/1yJL9Dj36eQm3vJ72GR2R6XuNFcDh6+z2+u1rNiiPbwXhtAPqjXc+o8PBJULc8a5sJdTtmZP3/9xxNR1aoXYUQdSEs3H0k0F3RoL1UjlI4kF
RDxP9IvaOIpkHE9QIlUlMpHhvlqUPHMbt0lDtl+8GzW3xgIkJXhh4CVOEXa/xE5QBy2J4IIQ3zLgibo+W8j1Nm2RgTcYjvsOWQ7k7P/c7Q4eY6HZ2/mFhHz/FYHOGBXj+V/tzq5LPntSxOKsg0Da8Z9oFp6g/3HRQ+/eEnO0q5H/kE7H9xuCzGR0
LBK2y+Bw80qFa/oJpGK9kJE0kh8xV8D4VR5vgjwkqhXP3v78bLNZb/EUEjDoyNLYsbqW0r0D5l33vGH07yvk/eBXZNQ5kwqxWjJz1kuZ2c0R5uACcwRyr1Q0bSgDQHA/XPwBbUTFRQ6x/b3dI4jZ/rH6KCEpZ1GA/tuaBqrU4v/3k3CbRrVXI0MWVz/Wpa+Hc4uP5H6jxzjUZPZJMKRe5FIKSUrN5uphaH6wPpn+3Uk8dTFBPdxFh4b4V7rY1oo8hmX1yfHvajkdHyRSsWLTAQe71aaQ0HwQgr1b/Kx9hfr8tvFf0Z2uBUxmvGLHaZhKMFEfKwT/v3fMsRxlJv/IsMJfXHt3SXw48xfy/YBNRmXpa7e76F9tjbXzo+eujHmJBv1LRGTF7Q3A0r6whnkZ8s1oaZLDaTBaw29ysQFLfSOp7V4BvPWYkdNBpNfTsafYyhpaAaazMfffirG0mw0eFuhyrrQY0eeWR4ZSWWkmbndpThVm5JaGGOnzq3ZIlXbAzmVhN9WpmDqZQKEvrBc1VeGUzAapEPvTw13LlGXUwjF1cS4LFGS7nymBGRWRhJvnf2kVFZ2oZi9PZxLN0Y2txWk6UeF1ZFhn6TXfM2gHqEiL8eDNF8d7IfQSTKRDdi3o7XX4pLbW2fXxzgf21OJdjgRN1iYnXaJ7KMgNchgqkN+iJgf08A2j

a3/xDOJqdZyTaYQ4jTJtZLZrrRt9ZEN4NqXUB0Ca1vpRd8cihDidfN4LcToZSRanOL6a6aqUsiilJgJTgNlu
Dku0cEqpeOAyzEi2EKIZkM97IWomq/0IV20xXzO1wXwddofWer17QxItmVLqSeA+4Fmt9V53xyOEOE
U+74WogZRbCCGEh3GMDP4D01/1Pa31cxX2z8DUJjpn4L+utX7Pse8GzKILAE9prf/TJEELIUQzI0myE
EJ4EEfP2J2YCZ2pwGpgmtZ6q8sxM4AkrfVdFc6NwHQ9SMJMtFwLDNZaH2+a6IUQovmQmmQhhPAs
QzErraXosqXBp9Ty3POBH7TWWY7E+AdgYiPFKYQQzVqzq0mOjIzU8fd/DCCHqZe3atce01lFuD
KE95VdPS8UsD17R5UqpMZhR5/scrfoqO7d9JeeilLoVuBUGMDBwcK9evRogdCGEaFrVfWY3u5uQQ5Pj6e
NWvWuDsMIYQSoF6XU/pqPcrtvgU+11kVKqNqdswK5xNqMsFtNbvYBZ7ICkpScvnthCiJaruM1vKLYQQwr
McovySu3FUWCJXa52ptXYuHPEeMLi25nwohRGshSbIQQnkJrDHhKQMn8hJVuXmDRQUpRlQ8eaUUlJSjfdYsWJFg8Ta0slIshBC
CCFECzZjxguxuv/12hg0bxoMPPsiqVasYMWIEAwcOZOTIkezYsQMoP7Jr+cIpIXkxCzUq8NVrruOuOqfZmEbN4ejUu0gQoaTOmeOxLBl3wVbfTH2+d8RJHnvzBHmO5jZVYyB8tcSC3fgSvYDyh6n
CCFECzZjxguxuv/12hg0bxoMPPsiqVasYMWIEAwcOZOTIkezYsQMoP7L7+OOPc9NNNzFu3Di6dOnCa
6+9dup6QUFBp44fNM4fN24cV1xxxxBb169eLaa69Faw3AvHnz6NWrF4MHDz+6NWrF4MHD+aee+9+SmJiIn379WhxBBCiHr4v2+T2ZqW26
x4CwGazMWPGDDr27UtiYiKvPIKAK+99hozqCQn069eParbbWjqQkWQhhBBCiHr4v2+T2ZqW26
DXTIgN4W8X9anzeampqaxYsQKr1Upubi7Lli3D29uba/rbbtvQhIGuzLFBXlJi7tLli3Dy8uL3br+TKLo2NRT2FhOoX4kKCeTfz+BD8hw32MVpgz
SOO+44rUXa+vXrSU5OJGhGjjRrF8+XKKSkpM9knfjJjHzVrKSiUkRTM1fm+WDa5cTKCJ1WvPWb1ZqI2jjAVqVN3tZT0K2VRu2SSgSXgSXtaLEKjuYtSs9uIV8+pCwo5yeMYkyaTdWGM1YT9m1dGzxOldjX4SWMR2Xj
nnncfs2bPp0KEDhw4dYsuWLQBkZ2cD8fX1PbWsqMpIshGjxikpt7D56wt1hiFpKzy3kp+3p
5BeXujsUITzG1KlTTsVqtAOTk5DB16lT69u3u3LfffdR3JycqXnTJo0CV9fXXyjjI4mOjiY9Pz9PbXyYOHUcXB
wWi4UBAwawb98+2tLkrx69Wrg5Gjt4hGRHVQFQXUl5eXHvttSxdupQkzmLc3XffzfbfZ58w
kJCQGgX79+XHvttXz00UfK0eKe/8/sIv3lqWw/OEJxIT4uTscYUe45x2x44cS
9DouWqz4hvYwkMDDz1+K9//ssJv5Rw/wQ8nHxS2Eoq/M1fBtpz5r8e/X9sOjmcDSM1Ezu2//3X3qqxe2Zx5MtPSM8fZ/OFhgwXSc6JXdvurr6F1l8JzX9qL6GPqbVs3oUZz7oyXVrg2oA6/vRM8hoH4iTb3WwI3yfjgajwcm0gaVv0ydkbBQzJCtRjXr6k5o5ioSQq2VYH
LFizg7bff5vPPP+eDDz5gM2fPs5g7ty5eLFy+GyAdWrMvX/HmRUpPvwzUedmWb0UBGP9I9YI9vVfc+11Av6UpH7cVwyqWhXKxu2GR6y2UlA0K9ybCZUehR5NOHWMJGuhnnDBJmJCNt0dd0+bRz/ho4jbADg47MkGqGAwb/ztpfPzmFOFEp80Bk+4LPNGvh3DyOJpFG/5AW
2Phs9QFK7Zp5mw83yj1Ew/JxjHYVl9rdHIkqniknJ4f27dsDMHP+zQ+OOPc9NNNzFu3Di6dOnCa
jQofz8888cO3YMm83Gp59+EYncmx2tFm4dcTYkYePmPmfM0CsACRORpaZvXa+0PmdWUKc1Zx3aePa7r33mq4yW6u8G7c1PIBdsEPGOfwbQKw0S/+JGFoQbJeYbYmC+Dt/AVrHT
3MGCvGoNX0tQZr7bD+hWs39SNFHyTJpDYvYxp3Jp+DVPP8/9N6PR1BVk6a0BLAqCMg2MOg00o3T
WRf8ILIRpFUamNh77cxOwNaQBkF5Tw9vTB+Hg17L/Nv99ymOP5JQT5ejF302FuHFX71ZSEezj/G5Ak
WYjG8eCDD3LddfMtRvVM+Nz79CDzO9/Tfw4R5vHxGmafAa/hcnw8Av/Dd0dVq6AvCjV37sbGdT0xL4+hsLmEQZAkBBV3TKKfUFTOI8VoEkB2Ab3k+TTaY5Iu1xNR/xcw/nr0i4ByOYJbhnKbqAQ+ne2HHW17X1RkgyTYfxd5qB7/PT4Izg4Pmbg9XyIwD2
nnmP8+PForZk0aRJTpkxh48aYwJQ+t3aZBjJJ6kXeFAAAhddLf0IzvvBE1y1L5Gvd2zq5ohb8ncPA1pE5AhRI/l85xmBNIaQZFw6rjMYlkVOM/3WQvmPa4cz4Ya/HIRTe+NwsPRJGhtazVv3hiyb9DmzTMu8m3tRpGvRw5RHvd/ND
9ascXcYQogzkFNQwm0fruG3Cz+dH5PQv29eXT2FiYltuO1aQOxWhquEf/lb60gM6+IywbF8fIPO/n1zxN
oF+pf5fF7j53kgS828upVA+gQEdBgcTgppdZqrWvuseRB6vq5vWTHUWb8ezVf3TGSwZ3CGzEyIRretm
3b6N27t7vDZLu8vDyCgoLQWnPnnXXfSt37rvPneHVa3K/uyq+8yu1ZCOUmqiUmqHUmq3urhSvZ3
VEotVkqtV0ptnrdSvjqmRJuR4CXtZqHJ5Um3zBtHSVFqYRcudwfudctJdvFDAvaVXvvrmEetB3YCtdrfV4cmC3Yq2tfqc+fftW7r+qO5AEIEzZCCGBPbiyQ3PzMbH1JP3h8h1eJT+ia3yY2RXhTNH9tRMgBsPAg2lNaxo7FmsZ2I12JapDT4/6r6oUFg61gjSDzsAsfM1tF91I
3H+bBLzdhtzfMP863Hc5l7f7jXDusExf1jwVg7qbqSy5mrT7A2v3HefmHneW2v/zDTh7+ahPNbeDAE8lIs
hAt37vvvsuAAQPo06cPOk06cPOTk53Hbbbe4OqcHVmCQrpazAG8AFQAIwTSmVUOGwR4HPtdPdYDgauBNx3n
egEfAbdrrfsAY4DSJgteCCGEEKKRPP/88/Tt25e+fftyn94ddZ1y5dy5cx+mmk4OWfnLB6Pp2O4FuMQ6r89FmkY3vi6X/r5TryrDqmNM6IMKhzWN
n/+Qk5+7T4mPl65Hx8vC1VqMzYCB9YkOjqZCXOOYW0Gr3syeW3RLZ1FTr+uhz4JrY2azcsmYvyLRP1stcqE
Yfq+KpFXFk6k2SbJMlCtFT33XcfmXcfmHcfm3xoXevNkYxOz99ccccBDT8N3PuVpuR5KHAbq11ita6GJgDQK
HI9DgTTH4/OATVrrjQBa60ytq0S6MOP+nbsOLE1LWNO21NvH8eEW0KB30U1XWp+1aFBvaQ3d1fv1ipzgVmf7W/Qg
8O8Oz328slyna75rn529l8KIdXF+087yK8pe0xkVaOZzq4rpD+xIB6eCsqpHhbDPKQ1CrIvqYQDW28/DQ
xF4E+Xjx94U7KCyx8g3mm+kYEcCADmE8/sk5Wv7SrODPeVvNXT4mMJAshmrHaJMntgyYMuz1M

d21w9DkxXSqUC84C7Hdt7AFoptUAptU4p9eAZxiuEaEK/pWQy4e9LOJZXVOn+rJPFnP3yz4x87idunL
maFXsyefbSRKYMqPgRYSilePiCXlw3vBPvLE3hg+X7Tu1buDWd3Ufz6BoVyIe/7mf30epnMC/ZcZSTx
TauHtLx1LbJiabkoqouF3M3peFtVVyZ1IHfndWFBcnp3PPpevYeO8kzlyby/OX9OFFYwlPfba323uLM+M
hIshCiBWioaebTgJla6zjgQuBDpZQF0z1jNHCt4/elSqmzK56slLpVKbVGKbUmIyOjOjgUISQSQpyp+VuOk
ks/XHDxtn9aaB7/cRGpWAX+7KIEvbh/Bur+ey9VDO1ZypTJKKf7v4j6cmxDDc99vY8dhE8j2dM2GSTm6w
mgE9vGY6/t5Wn5lafqC7fnUmwrxeDOoad2taxTQD94kJ5Zt52Ov95Lp3/PJdHvtmM1vpUqcVZ3aMIDfD
mptHxhAd4s3BrOpcPimN090h6tg3m9rFd+Xr9IZbulM+ixuJjlZpkIUTzV5sk+RDQweV5nGObq5uBzwG0
1r8CfkAkZtR5qdb6mNY6HzPKjiDbTW72itk7TWSVFRUXV/FUKIRrFmfxYAn6w8gK3CZLuPVx7gx
23pPDixJzeO6syQ+AhC/b0ru8xpLBbFC5f3IyLQh3tnreHrelsSs3hjrFdiQ7x456zu7NkRwaLdxyt8hor9hxj
WJcIvKzlP8aeviSReyZ04+7x3bi4fyyfrDzAO0tTWHcgm7ScQib3awdAsJ83f76gN73aBvOXSWWzne8c34
0uUYF8v0V6LjcWmbgnhGgJapMkrwa6K6U6V8MBPz5lQ45gBwNoBSqjcmSc4AFgCJSqkAxyS+sYB
8jylEC3CyqJRth0/QMyaY1OMF5UZWdx89wVNztzKmRxQ31bMvcXigDy9fOYCUYye565P1tA3x49JBp
kzjhpHxdI4M5J5P1vPwV5tYsftYuSQ99Xg++zPzGdk18rTrJsaF8sfzevLH83ry6lUDmJTYjufmb+fpuVvxs
Vo4JyHm1LFXDunA/D+MIcJR0wzg523l89tG8MylifV6XaJmp0aSpdxCiDobP348CxYuYsKLft1Vdf5Y477qj
ynHHjxuFs03jhhReSnZ192jGPP/44L730UrX3GMdoq/ckiVLmDx58hlfp6BRrUuBuzAJ7zAA2HszGZtc8c
RrrUuBuzAJ7zZMF4tkpdQTSqmLHYfdD9yilNoIfArM0MI0BWKe1nts6rIr0MI0cA2HszGZtc8c
H5PooJ9+ei3/QBk5hVx+0frCPTx4qWp/bCcQc/jUd0iuXVMF4ptdm4d0wVfL7MSm4+XhXeuG8w5CTF8
uzGNa95byaOzt5w6b8WezFPnV0cpxYtT+9G7bQjrDmQzpkcUcUIX41j3ZHBvmiVMP1chblyUiyEPU3bdo0
Zs2aVW7brFmzmDZtWq3Onzt3LqGhofW695QmJ86d+ql+vmrWD/MHTu1+Tnk7jBDvaLKptG0tWbkcqnzTg9w+N/MVE5yeBGoAQdbY+WwG8zdQmt9P9CPnb0hnW/Ksn57hEsQC/dZ3Z+cpD
Hm/VWo/SWvfXWg/QWi90OfcjrXUXUfrXVfrbVM3BOihViz/zhKwdDOEVw9pAM/7TjK1rRcrv9gFQez8n
nj2kFEB5952/MHzuvJv2cM4foRncpt7x4TzCtXDWDXQrV9as5khOIQArdh8jMsiHHHjFBp18wcw+k
rT/1NMDHi3dvSGJwp3BuGh1/xvGKMycT94SovyuuuIK5SdX+dSXGy68Ozbt4+0tDTOOuss7DPOnVn86kIeGFm7DzGajIbLCjkDcTTPSukDpKsUmuk
w9/+9vfKj0/Pj6eY8eOAfD000/Tq1cvzjnnHH7x1Zdn7nR+dGaqPU5+DPTDWpYO38ZJ1O2iiBBe5IEPNntqHP9r6LeojWzFH3YC/4Oro9oOxF5OZ8FqGO6E4RP0K/4/Qx+vDI/bzyNqVUS+hTHl+m/Xn6RHxkSSYkRrmzXrW9XmTrZfnD4gbzY2yZPkJdu+dXPyPe+//ZjN7wIu3bYEUHgt1sh+MaXUtvlzP9d99Oc9oDlqWCUUGxNVGo2LwNYoLuHqR7d2HJXZQrC2KB6m8/IdO2rXt5p7j8O6GsNjMGlewxcoMTjGLJnhk2rZGK/2RqS2pWH7sXwCVhgMnmIBwQ/0TqVG/t/hjdB9u4/tEgDh4Ci/1iY7Ak6xpVoPcjnyHP2U8/p2LVIvfUzXbwSxhy/HrVjxFh9fJoMl3i8vfwxgUPyw/QMSXj25dqtzdXGadKtnCIjC2HTAX7lAsSBBVEGRwxBTlDtNE0fFNjVhDbPvYTZd1X6kO9eeXxrdj3QQSJNBTCqnXyahYC7vRCpyi9M9KVk+dtSfpq/6UeYwVIfBxEPeXERdtg/ffeEIt3dZrYiKJxeD/qWSA0ws/bwxuVelnfl0TSK8QQCn5YHWjBLY6Xu6VNI7Etcq6GI2GKaBqXq2DKTpHFWVtmEqx+cU0Z+F2CN5ECDfPODX0X4flczpo7n58ASzQtBQA6dHcJTlp4bNpWKKP2AyodHKQr2tfnp0uq7Lbn9B9UZu77wJWF8HG3/Uemp8b57RSyKzPb7LORUAyrALoQADo/+VwfwoLR91ixjiCKvOuPyx/hHnZ5IYaEiL0RpeYlwtArgd+KnkLNmoVdhBZXbF9Vw0UpCeXfmKm/NPOn4T8nY9b+zlatPQrOxxqODoYXZ/wMu5RWoYfgRrYGeqMbeJdXs1ezEUi9fSfSpLB/GK4xYqPN0rCFfoIDZFUzSkJFE8NiXpHBsk+YQMG1gG4jafVfsUZv0OBoHmVJB6B8kRWPfE6tBvuwe67cYKwoETJjWcflw5Wl0szkbKuOGPMrhvaLTN4nsgPFBmXc5uJ7eOlJRAcKEoG9OXFLWKsQR6W5ExJnLLr5WwB9tc2uqVdf1Gw1jQuXbD9ZxNfOKc8jfMrMnSF+GSd+vZrdHCb/IcEF6cVU8m4oH5B9GcKq6bR5jPGOJTZyMlpxTnOoRKNvwhzMQB7QpaIgkZPLtC8SKU2KThGeNuJ+1Prf3VXK+LzqBPmtOmKAfMn9v5vTmNRYhhCQzZlwoQnfZrU/9cnhKDP3rHHx+6oXHlJ/fvwMBWb5mrhzjh0ASQw6c/BzsnGJVQ9hnuwnHqgNBSNkdGRO4yXbXwIb23Xqc8a06qV53BRqP79aPxVDa1HeDt4LzqCcuFh3PZTqUT9fgfqD8Z8Wq4SqlHrBkqlHLXGcHPYbmjP5q+2BNLRfv0hrn6ezV3tOgxMBbT9WwJtG7Bz/3hgklVhDpycXBZ7kqB7/kPLOxuGVWOdR7hr9jfgbT2e2D3xSbk5tyRM5u+nDtNPTycMp8kFWaW7/dQn6vb1RVjuT3v/4DPRubHcX+uJ8+HcSVnE2l4bdxRQrFcbrtP51NiDfWMzN0pmDBaWQMZFLt6YY3Y/1g7jB0YfDJ0dOHkmZ1Gd4kRWELa3pvHmlfvWmrt+0Y7yKCP9cv9T4j41Sy+lrKvnT5dBXaItbMXVcKwm3ukSGZWCh2NrQgs0c7pWvbJY4r0kH9YAzqjdZOBjdK7oT8O5K8iZdV7Kmr4NsfAxhB5QbYeTT5mN5zYRMd7sBaOJeUsYTKTM/wHs9nGZwSMcxcmJUHNLDTZA4E1Hc2n5tZFZ4YjtrfP/HtcRmnbQzWQ+B2rFuT5mh9Lwg+TR7m8hPkdONPU0y8yNhRl1lA6WAJkLBXXnbtVw1DzHHX6IRuRtMkhZHSDBTnE9eX6EgGnq4K6OCZtVM0i0MhqpNTyYc0M0uYmY9fW09wd8HHnZK5DHQOgNDBPjvlbsVhz4xfDtfz2vPfB3tt1fCBE9zGFzFH+h3wJjz4PBpnSMH6I1t3c7tVjC+zbkdZAzkE3DEn4N1TpCfmYjvsx9k8Ti+Vv3G/v8XFdlj7MpTQ3K4iuChCCsiPUQAgPSb/wIDVb0+xpzGNj2fRTdL+LVHDAfSP0P8tIi4Mz3zm/HTN2zYc0l6r6M7aJEP8jFKsaH9jc1n4e3kxFj3lYfQKkxbQKN+UlQtYCCIQ/4mDIB5J2Ofz3FzFJVMyL+qGVIaKc1DvXKvB+JDeL7c/CT86WqYXDlK6qqvmwXcG5XbNpJ3x+8Vx9M4qVRZ4fIqg0pQ5Ks3C34Nc+uEDQIfsWafbfVBfqc3s8vXBpF9Bo7eYcXyfznUTMfZ4xOj6jiwmS3nQaD0PrjT1Yp6V5xs7O4q2O6X8fW6Y9BlRnz8P+KIf9gqfQRF5Jf6c1RrUd1zRsSFSfE1KjSjPrdkKhN2WTxRWdL66xQ+aIg3RuXWqhHpTfTEHdsS/2Vb4pQaYIrAftHzN2PT3RA3eeKS+6bg+lxvhpQN7ZTZdcJWWLjPz+iTNIgS7D1S9ovz4fM3P0UnbI4aUvGRl+TvbmTO+uQ7+3PkLy+7Ntl9Dqd3S2n62WLZPH1U/zPXJ/4xYmGjHk7M3i7h3V3tSM8s8vknXSsnV6dhJhXt7mcVS0m1+iOa4nsRfFmZaDYWcprOdzlt/JtLlfG8E2ny8R6tPyujKSQNNT1+Yd98dkTA+T7bJuFGKn4rsXZNOHMXbS6PtlRxQmKWkp5CA9VNLZ9

xLhQ/jIpge/vPYtSm537Zm3AZtckp+Xw2ZqDp1qjvfbjLtbuPw5AUnxZkhwa4F0+QQY4vheWvgDf/sEkzA
2ttAi2zwWfYCz2Yia0KyY5LZe4cH86tqliCVRnknx0W8PHIxqMj9UiI8lC1FNQUBDjx4/npptuOjWKnJub
S2BgIKGhoaSnp/P9999Xe40xY8Ywe/ZsCgoKOHHiBN9+++2pfSdOnKBdu3aUlJTw8ccfn9oeHBzMiROn
f0vXs2dP9u3bx+7duwH48MMPGTt27Bm9xqFDh/Lzzz9z7NgxbDYbn376KWPHjuXYsWPY7XYuv/xynn
rqKdatW4fdbufgwYOMHz+e559/npycHPLyql+QqjYkSRaiEX3wy14yThSxIPkIdnvNSeTO9BP8YdZ6+sa
G8tLU/hSW2Fm77/ip/XM2pjH6+cUs333s1LaPV+7H18vCoj+O5bYxXZi1+gCvL95N+3B/Xri8H/+5aWi5/
sVdo4J4YkpfVu3L4s3Fu3ni262E+Xvz9CWJXDOsI1+uS+WbdYeIDPKlY0QViajTvl/M7yObTDLb0Pb8B
EW5MORmAG7sZZKqUVWNItttUOB4vzJqPzlEND0fLxlJFuJMTJs2jY0bN55Kkvv378/AgQPp1asX11xz
DaNGjar2/EGDBnHVVVfRv39/LrjgAoYMGXJq35NPPsmwYcMYMGXJTgwIH
s2bPn1HY/Pz/+/e9/M3XqVBITE7FYLNx+++11ej2LFi0iLi7u1M++fft47rnnGD9+PP3792fw4MFMmTKF
Q4cOMW7cOAYMGMD06dN59tlnsdlsTJ8+ncTERAYOHMg9999IUQnROLZt20bv3r1rPlA0O5X92
VX3mS0jyUI0kn8tTSGvuJS3pg/GalH8uDW92uPfXpLCrqN5vHB5P9qF+hPo68WgjuEs22UW8UjPLWT5
7mP07xDGjvQTfLr6ILPXH+JksY3pw8uWgu7UJrDaBBlMTehTl/QlLtyfXm2DmTbELKoZFezLjaPiARjc
qeqEHjDlFfuWQfxoGPMgpG+B7d9Wf05dlBTAjnnQ+yII6wjeAXhl7+XvV/avPEGGslILLz8ZSW7mfLyk
3EII0bxJkixEIzh6opCZy/dxcf9YhnaOYEh8OIu2Vb3E8u6jebyxeDcX9Y8t1w5tTI8oktNyOZZXxLcb07Bre
PnK/gzrHMHLC3cwc8U++sSGVJ00ViPU35vv7z2LL+8YWW5p59vGduXKpDgm929X/gRbafmOEZl74
MRhiD8LEq+ANt1h8bOQsROO7Sr/U5hb5/jYvQiK86DvZaAURHQx96yOM0mOGwK5h6Awp/LjCo5Xv
t0pYwcUZNc55OZCKTVRKbVDKbVbKfVwNcddrpTSSqkkx3NvpdR/lFKblVLblFJ/bqwYva0WSiRJFkI
0Y5IkC3EGtNas3pdFxbKlf/2cQrHNzn3nmHY45/SOYUf6idNar4Fpt/bI15vx97Hy2OSEcvvO6m5qb5fvPsb
sDYfoFxdK16ggHrsogeyCEnYfzWP68E71Xh0u2M+bIN/ys39D/b154Yr+tAv1L3/wj3+D14dC8UnzfN8y8
7vzGLBYYdzDkLEN3hgCryeV/3nv7LoHl/w1BLSB+DHmeUQXyKplkhw/2vw+WslocsoSeKEL7F1W9X
X+OwXmN1p+2KiUUlbgDeACIAGYppRKqOS4YOBeYKXL5qmAr9Y6ERgM3KaUim+MOKW7hRCiu
ZMkWYgzsHBrOlPf/pWfd2ac2maza/63IY3z+8QQHxkImCQZ4Mdt5UsuSmx2np+/nVX7svjLpN5EBfuW
298nNpTwAG9mrtjHlkO5XDKg/ant1wztSESgDxf3j23Ml1hm5wI4kQarHU3g9y2DoLbQppt53vdymDYL
Ln+//M+A6XBsJ+RlVH3tiorzYcd86H0xWB1JfJuucHyfGdGuijNJ7uSYsJJRSYeLjbNA2+GnpyrvyFF80oy
Qt+lS+3ibl6HAbq11ita6GJgFTKnkuCeB5wHXHlEaCFRKeQH+QDFQj68BaubjZaFIkmTRQjW3+VyiZv
X5M5MkWYgz4Owy8e3Gw6e2rTtwnGN5RUzsW1auEB8ZSNeowHIlF3sy8rjirRX8a2kKVyV1YOrguNO
ub7UoRnaLZP2BbKwWxUUuCfETU/qy+IFxBPo2Qbvz3MOQuQusPrD8H6bsYq+jHtk5iq0U9LzAlF64/g
wwM685vKH299v9A5SchD6Xlm2L6Ar2Usg5YJ7b7bD7x/KJrjNJjh0A3gGnjyQ7W8oFRsHB38ykw4qyU
sru1zK1Bw66PE91bDtFKTUI6KC1rtiS5EvgJHAYOAC8pLWu3Yo2deQj5RaihfLz8yMzM1MS5RZEa01
mZiZ+ftXP16lIFhMR4gysTDH5w8KtRygq7Yuvl5UFW47gY7UwvmdUuWPP6R3DB8v38sPWdBYmH+
HbTWn4eVt545pBTOrXrrLLAzCmeyRzNx1mVLfIciPNVosq19qtUTlbvZ3/DMx7AOY9CCePQuezaj63b
T9AQdp66H5u7e635WuTyHZyaWHUxpG0ZqaY0out38CXN8GMeRDvOC4/00za8wmCqJ6njyTvXmRay
l36iXkNi5+BLuPLEn0oq3tu02KT5GoppSzAy8CMSnYPBWxALBAOLFNK/ai1TqnkOrcCtwJ07Nix4u4a
+XhZyC8+s0UFhHCHuLg4UlNTyciow7djwu38/PyIizt9MKo6kiQLUU+ZeUXsSD/B0M4RrNqbxbKdxzi7
dzTzk48wqlsbgv3KJ7Bn947hX0tTuOW/awj29eKifrE8cH5PYmroRDGuZzSBPlauHVb3RKTB7FsGvqGQ
dJMpu9j4idkeX4sk2S8EIrtD2oba3av4pLnHgGvKSi2gbGQ3aw9wjkmkwZRgnEqSs0wds1IQ1dv0WXaV/
A34hUH382DMUfjuDyZx7n5O2THOuueIFltucQjo4PI8zrHNKRjoCyxx1LK3BeYopS4GrgHma61LgKNK

qeVAEnBakqy1fgd4B0wLuLoG6eNlIbtARpJFy+Pt7U3nzp3dHYZoAlJuIUQ9rdprRpHvP7cHof7ezN18m
OS0XFKPFzCxb9vTjk/qFM5fLuzN29MHsfrRc3hxav8aE2SAmBA/Nv7tPM7vc/o1m8y+ZdBppGOCnmN
CW0j72ieS7QaYkeTa2LkASgtMVwtXQdFmhDhzj+mWsesHsz3XJf/Lz4SACPM4uhfkHSnrZOHaUs7qD
QOuhdCOsOSZ8iUbmSkQFAO+wbWLt/lZDXRXSnVWSvkAVwNznDu11jla60itdbzWOh74DbhYa70G
U2IxAUApFQgMBxqll563VVFSKl9XCyGaL0mShajgYFY+n646wMmi6r8K/i0lE39vK4M6hTOxT1t+2Jr
OtxvTsKiyiXquLBbFLWO6MLFvO/y8rXWKybVFW5PLOWTqdJ2lFXGDYehtMPSW8mUK1YkdaCb9n
ThS87HJ35gkteOI8tudbeCy9sCO78FWBCjISS07xjmSDBDtaOjgrEt2tpRz1jl7+cCwW+HQWjNRzykrpSX
XI6O1LgXuAhYA24DPtdbJSqknHKPF1XkDCFJKJWOS7X9rrTc1Rpw+XlbpkyyEaNak3EIIF+sPHOd3/1
lD5sli/r5wB3eN78a0YR3x9To9qV25N4uk+HC8rRYm9WvHZ2sO8u/l+xgSH0GbIN9Krt5COeuRnW3VA
C58oW7XiB1ofqdtgJ4Tqz6u6IRZYW/Q9WbUuqI2XeHwRpNIh8SZUePctLL9+ZkQ5qg0iHIspZqxDTqN
MC3l/COg89iy4zsMc8S1HkIckyKz9tS+drqZ0lrPA+ZV2PZYFceOc3mch2kD1+ikBZwQormTkWQhHBY
kH2Hau78R6OvFm9cOolt0EI9/u5XzX1nKugPlF5/IOlnM9iMnGN7FjFqO6NqG8ABvim32Skstmq3Fz8Ivr
5bflp8FH0w0nSMA9i01dbwxifW/T9tEUJbKO1xk7YU3R8I/+sPrQ6C0sHxXC1cRXeH4ftizCPpcAqEdTi+
38HeUW4TGgU8w/Pi4ufbW/0HCxeXrnGP6mric9dJFJyAvvUWPJLcU0gJOCNHcSZIsBLB4+1Fu/2gtvdq
G8PXvR3JhYjs+vWU4/71pKCU2zdS3f+XlH3aealm1aq9pNTa8i0nIvK2WU8nxee6sHa4Luw1+exPWf1h+
+8FVcOBX+HQa7FxY1urNcgYfF75BENmj8rrkJc+ZEocOw8zCJKP+AB2GV36dNl1B28BWbBLp0Pam
HARM/+TC7LJyC6Xg3P+DHhPNtftdBSPuLn89nwAzwc8Zl7P9m4d2tmhOfKxKWsAJIZo1KbcQrU6JzY
6XRZ1ape7oiUIe+GIjPWOC+fSW4fj7mK/5lVKM6RHF9384i8f/l8xri3bx846jvHLVAH5LycLf20pi+7BT1
73/vJ6cl9CW9mH+ld3W/UqLTR2u0+GNph1acR6UFIK3YxKhs21aZA+YdQ3YS2D478/8/rEDTbcJrctqm
Y/tgs2fw4g74bynar6Gc4Q3tCO0H2xKQYpyzAhwiWNNDGeSDDDkZvNTU1w755u4nO3fZCS50fl4Sbm
FEKJ5k5Fk0arY7Jopry/ngn8sY9vhXOx2zQNfbCKvqJTXpg08lSC7CvHz5uWrBvDGNYPYn5XPha8tY87
GNJLiw/HxKvtfKDLIl/G9opvy5dTe9nnwXEdTquDkXFZa281CIU5Ht0NwLMz4Dtr2Nds6jznzGGIHmlIG
10lyPz9v+hqPvLd212jTDVCm1EIp02EDTF2ycyERZ3eLWsc1APKPmQmALb/9W4vh42WRiXtCiGZNk
mTRqny/5TBbD+dyICufKa8v59YP17J0ZwaPTk6gR0z1Lb8m9WvHgj+MYUh8BFknixnRtU21xzcr6z8yb
dWSvy7btneZWZUOyq9Ml7HNtE/zD4cbvoUbv4eYhDOPwXXyHkDGDtj8JQy9FYKiqjytnMA2cMMcG
PugeR7qSJJzUl2S5Dr+uZyKa71p/xYca8owRKPytlqw2TU2u7SBE0I0T5Iki1ZDa80bi/fQJSqQn/80nnE9o/
hxWzrn9I5hei0X6ogJ8eO/Nw3lk98N46ZRzbSZfHoyFOaUPS/MMcs8Q9kCHLYSU3fc9zKweJWVWNjtk
LHT1OmC6RXcaWTDxOWcJLflSxPHgkfAJxBG3lO363QeU9bD+NRI8qH6J8kxfcx7cHiDGUmWeuQm
4fwWRuqShRDNldQki1ZjyY4Mth3O5cUr+hEV7Mu/rhvMqr1ZJMaFnqpPrg2lFCO7RTZipGfAbjedKTqP
gas/Ntt2fG8muiVcAltnm7rbguOmFrnr2XBwddlIcvY+M+Ic3avhY/MJgNhBsOUr8wMw5k9mdLi+gttheiU
fMhMRoe5Jsrc/RDsm72XugV6T6h+PqDUfR+/volJ7nfuGCyFEU5AkWbQKWmteX7yb9mH+XDLQjD4q
pRjWpQWVTNRG7iEzGW/7d6asIXaA6Skc2sFMjNs62zxXji+R4kebhPjIZvPcmSxHN0B5RWWun13Wj
UJZznzU1svHrMSXm2pW0YO61ySDKbnY8rX5h4OMJDcJGUkWQjR3Um4hPF6pzc4PW9NZu/84t43tgr
c7V6+ri42flSWvteWceAamtVrBcbPSXMIUs8hGh2EmSd63zCy2ERRtSiuy9kJxPhzdas6N6tlwr8OVb7BJy
qN7QVSPyhcMqauQ9o6Je1ngHWhGhusqdqBJkEE6WzQR50iydLgQQjRXMpIsPEpmXhH3f7GR/CLz1X
teUSm7M/IoLrUTHezLlUkd3BxhLeUdhdm3m9Xhrp9d+/OcLcwG3whr/w0/PW1auPW5zGzvcxnMfwgyv
GHwDLMtuheg4dhOyNhuRp19q5/E2KyEtjd11IHRdS+1cGo3oOyxjCQ3CedIsiTJQojmqoUMqQlRO//bk
MaSHRkoBVaLIjrElxtGdOKlqf359u7RLaf2cdsc05pt71I4eaz252WlmJZq5/6f6U6x+l1HT+FBZn/CxYAyi
XPns8w25yS9jO2m3CKqEeqRG1NIXNnEvfqUWoBj8p43oCC8mU7I9DCnkmQptxBCNFMykiyazM70E/
x94Q4evqA3nSMDG+UeC5KP0DMmmM9uG9Eo128yW74xSW7Bcdj2LSTdWLvzMveYHr9+oTDyblj0
RFlPYYCQWOg4Ag6sgE6jzbY2XU2CmL7FjCZ3HdcYr6jxhMSaUonjeyGsdl1KTuPlaxLl/MyyRVVEo/K
WcgshRDMnI8miyXyy8gALktO57M3lrN2fVa9rzF5/iPs+20BOQclp+zLzili9L4vz+7aQZaGrcuII7F8OQ28

zi2e49jauSdaesoUwht4GSTebPsSuJvwFzn6srKuE1Rsiuzu6YBSVjSy3FM5eyZl76l9uAab38riHGyYmUS
MZSRZCNHeSJIsms2xXBv3iQgkL8GHauyv5fvPhmk+q4J8/7eKb9YeY+vYKDmUXlNv347Z07BrO7xPT
UCG7x9Y5gIY+l5qffb+YGuWa2G1wfF9ZTa1vEEx+2UzYcxU/Gs66v/y2qF6Quds8boz2b40pJM7xQJ9Zk
txrEgyc3iAhiZr5ykiyEKKZkyRZNIlD2QXsyTjJxf1j+eqOkfSNDeHuT9fz657MU8dknCjiqn/9yshnFzHy2U
WMf2kJO46cOLV/99ET7Mk4yaUD23M4p5BL31hOclrZohnztxyhQ4Q/Ce1CmvS1Nbjkr00LtuheZqKdtps
a5ZrkHDT9kOvTnSHaZfS4pdUkO0eS4cySZNGkvKUFnBCimZMkWTSJX3ZlADCmRxQRgT7MvGkon
doE8PuP13IwK5/jJ4uZ/t5KNqXmMLJbJKO6RXIkp5D3f0k5dY0FyekAPDSxF1/dMRIvi+KGD1aTll3AicI
Slu/O5PyEtnVaGKTZyU0zK+H1udQ8j+4NkT0heXbN5zo7W9SnO4MzMQ7rZFbBa0mC2pb1fa7vxD3R5
KQFnBCiuZMkuZXRWvP95sPkF5c2yPU+XXWAS99czqbU7GqPW7rrGDEhvnSPDgIgxM+b924Ygs2uu
eW/a7jh36vYm3mS925I4qWp/Xlxan8uGdieORvTTtUfz99yhAEdwmgb6kePmGD+e/NQCkts3PrhGr7fcoR
im52JLb0e2ZkMO5NkpcpKLrbOgV0/mhXyKpPl+AfFmYwkR7ewemQAq5dJlEFGklsQaQEnhGjuJEluZd
YdyOaOj9fx3rK9Z3wtu13z+k+7WX8gm8veXME/F+2itJKvTm12zfLdxxjdLarcKG/nyEBev2YQO9NPsD
Utl7enD2KUy3LP1w7rSGGJna/XpXIou4DNh3I4v09ZEtwtOph/XD2A5LRcHv1mC5FBvgzqGH7Gr8utdn
4P0X3MRDqnvpeb359fBx9fDu+fA3t+Ov3czD1mMY3gevxDIbwz+EdA+6T6xe1uzpILSZJbjFPdLaTcQgj
RTEkLuFbmx22mZGH2hkPcPaFbtaUJRaU2rnjrV6wWxcX9Y5ncrx3RIWXtsX5NyeRQdgFPTunDyr1Z/P
2Hnfzn1330ahtCz7bBTB/eic6RgSSn5ZCdX8KYHpGn3WNMjyjeuyEJP28rI7uW39+3fSgDOoTx8coD2L
XZVnFS3tm9Y/jT+T15Yf4Ozk2IwWJpwaUWdrtZSjpxavntUT3grtVQkA1o+PImWPwMdBlf1toNyjpb1Kf
cxOoFd64Ev7D6x+9OIe2B1ZIktyC+MpIshGjmZCS5lVm0LR0fq4WUjJNsPpRT7bGfrDzA5kM5nCgs4Yn
vtjLiuZ9YmHzk1P7P1xwkxM+LqUkdeP2aQbw9fRDjekaTU1DCh7/t58p//creYydZtssshuE6SuxqQq+Y0xJ
kp+nDO7H7aB5vLdlNz5hgukQFnXbMHWO78upVA7j/vB61fRuap+N7oSgXYgecvi+yO3QYAh2Gms4U
qavNctOuMvdAmy71v39QNHj51P98dwp1dLiQJLnFkBZwQojmTpLkVuRgVj470/O4fVxXfKwWZq9Pq/
LY/OJS3li8m+FdIvjxj2P58Y9j6RkTzF9mbyEnv4Sc/BK+33KESwa2P7WK3cS+7U6tbDf37tHY7Jpr3/2Nb
zem0Sc2hMgg3zrHPLlfO0L9vTmWV1xlazelFJcMbF+v6zcraevN79iB1R834Fqzit7ip0E7hthtpZC9v371yJ
4gfjS06w8Blf9jSzQ/MnFPCNHcSZLswbTW2Jx1CpSVWlw2sD0TekUzZ2NapTXEADNX7ONYXjF/Or8
nSim6RQfxwhX9yDpZzHPztzFnUxrFpXamDu5Q6fndY4L5701DySsqZfuRE5zVPaper8HP28oVg80o4Xl9
WvikvJqkrTdLStfUgs3LB8b+CdLWwc4FZlv2frCX1q+zhSfoeQHcttSUjYgWQVrACSGaO0mSPdTm1Bz
Oe2Upl721gqJSGwCLth2la1Qg8ZGBXDKwPcfyilju0qfYKaeghLeX7GFCr2gGdyprqdW3fSg3j+7Mp6sO8
vpPu+jVNpi+7avuSdy3fSgzbxpK3/YhXDIwtt6v5Q/ndOft6YPp2z603tdoEdLWQ9tEswJeTfpPg/B4+OlJK
D55Zp0thHADGUkWQjR3kiR7mFKbndd/2sWlby4n62QxGw9m89KCHZwoLGHl3kzO6W1KFsb3iiLEz4
v/rT902jVe/2kXuYWlldb43ndODzpE+JOeW8SVSR1q7Ek8qGM43919Fr3a1n+Bj2A/75bf2q0mdjsc3gjtBt
TueKs3nP8MHN0Kn1wFRzaZ7a11JFm0ON5W89khSbIQormSJNnDvL54Ny8t3MnEvm1ZdP9YrhveiXe
X7eWZedspsWnOdiTJvl5WJvVrx/zkI2w7nAuYlm5Pz93Ku8v2clVSB/rEnj5y6+9j5cUr+jOwYxiXDmx/2n
5RT5m7oTiv5npkV70mwaXvwP7l8NPT4BMEgfUraxGiqSml8LFaKLbpmg8WQgg3kAK+Fiw7v5hQf+9T
o7k2u+bTVQcY1zOK168ZBMBfJvXmt5RMPl11gPAAbwZ1DDt1/vUj4pm76TAXvraMi/rFUlxqZ37yEW
aMjOevkxOqvO/wLm345vejGvW1tTq1nbRXUb+pYLHAV7eYUeSWvNqgaHV8vCwykiyEaLZkJLmZO
ZJTWKvj/r18LwOf/IF/L993atvSXRmk5xZx9ZCyyXR+3lZemzYQH6uFCb1i8LKW/ZH3bhfCsgcncMfYrv
ywNZ0FW4/w18kJPH5xH6wtud9wS5S2HrwDILIebez6Xg7XfQMTn2/4uIRoRD5eFoptNneHIYQQlZKR5
Gbkx63p/O6/a5h54xDG9Yyu9Bi7XfP0vG28/8te/L2tvPbTLqYmxRHs580Xaw4SEejDhF7lW6X1bhfCd/e
MJjr49BZpoQHePDixFzeO6kx6bqHnT45rrtLWQ9t+9e/O0GVsw8YjRBPwtioZSRZCNFsyktyEcgpKOJZ
XVOX+T1YdAODFBTuw20+v08vOL+a2j9by/i97mTEynk9vHU52fgnv/7KXrJPF/LA1nUsGtD/VpN9Vj5
hgwgKqXigiKthXEmR3sdvMxLvKFhERwoP5eFkokZpkIUQzJUlyE5m/5QjjXlzMlf/6Fa1P/0shPbeQJTuO
0jMmmOS0XOa7rGwHsGxXBue/upTF24/ymKMkYkCHMCb2act7y/Yyc8U+SmyaK4fENdVLEg3l2E4oy

a97PbIQVVBKTVRK7VBK7VZKPVzNcZcrpbRSKsllWz+l1K9KqWSl1GallF9V558pH6vUJAshmi9Jkht
ZfnEpf/piI7d/tBaAlIyTbHV0k3D19bpD2DW8ce0gukcH8feFOyi12SkssfH4nGSue38VwX7ezL5zFDeN7nz
qvD+e14OTxaW8tmgX/eJCz6jVmnCTQ+vMb0mSRQNQSlmBN4ALgARgmlLqtJm4Sqlg4F5gpcs2L+Aj4
HatdR9gHFDSWLH6eFkpkiRZCNFMSZLcyP71cwpfrkvlrvHd+P7eMVgULEhOL3eM1pov1hxkSHw43a
KDuP+8HuzJOMlLC3cy+Z+/MHPFPmaMjOe7u0efVhLRIyaYSwaYVmxTB8socotzbBcsegKCY6FNN3d
HIzzDUGC31jpFa10MzAKmVHLck8DzgOts4fOATVrrjQBa60ytdaPNrPOxKllxTwjRbEmS3MgWJB9hS
HwED5zfk7ahfgyJj2DBlvKlFGv3Hyfl2EmmJpmuFOf3aUti+1De/nkPJwpL+O9NQ3n84j74eVsrvceDE3syf
XhHLh0kSXKLkrEDZk4CbYPrvgZL5X++QtRRe+Cgy/NUx7ZTlFKDgA5a67kVzu0BaKXUAqXUOqXU
g1XdRCl1q1JqjVJqTUZGRr0ClRZwQojmTLpbNKLU4/lsP3KCRy7sdWrb+X3a8sR3W9l77CSdIwMB+G
JNKgE+ViYltgNMk/3nLk9kzsY07hjbtdoJdwDtQv156pLExnshomp5R2Hen6DUMRgX1tG0YrO4/Ptz6Yv
Qtj/0OK9sW/YBmDnZPL7hO4gu+29EiMaklLIALwMzKtntBYwGhgD5wCKl1Fqt9aKAqt3wHeAUhK
SqrX7DsfLwuFJZIkCyGaJxlJbkQ/bT8KcGpIXeqxV3zpEv5Kny/oX
WOCLNxsx/ewdTbkpMLxfbDqHdj2v7L9B1fBT0/Bt/dAicu32oufgaJcmCEJsmhwh4AOLs/jHNucgoG+wB
Kl1D5gODDHMXkvFViqtT6mtc4H5gGDDGitQb6tFyi2EEM2WJMmnN6MdtR+kSGUiXqKBT29qH+ZPYP
pQFyUcoLLFx24drALh1TBd3hSnOxOEN4BsKt/8Cd6wwi4EseR7sjr/4lzwLXv5w4jCs+4/Zdmw3bPoMhv
wOonq6LXThsVYD3ZVSnZVSPsDVwBznTq11ttY6Umsdp7WOA1YCk7TWa4AFQF+lVIBjEt/ZwFYvx1+GJMlCiGZPJu41np+3H2Vo5wjG9GjNvM1pJRZvZaUnlz+2H+Vun2+OvPAVXTvHVyLlU0t7BJ3SQX+CvNJSOASaPNEnDPFnxEpUtiUltgzhBEKc5zYK1yElqtQZZfPRqW5+r+/Zh7gGG1Zg3YE4ieKYwUtmsRD+QYR3/1DmNXS/2FHbBrGEK0WIHRBJYe
B5AOF0KIZqk2mVx74KDL81THNlePA9OVUqmY3px3AyilgoCHgP+r7gZKqVuVUVUmuUUmsyMjJqGXr
zYrNr/v7DTrpFB3HJwIpvTwNJ2wALHzUT4g6thU2fw29v1yK4Uvj+QVj5VuPE5SpjG9iKIG4IxPTx4CR
5gxktD23F3xgIcSaCogksdowkS69kIUQz1FDDndOAmVrrOOBC4EOllAWWjkqPz9IUSGGx2Zme8/7fQBpnVLhcubZFix0
IaRvN5D1Pk7a+6kl7QoiaBUXjXyLlFkKI5qs23S0OAa7DZXGOba5uBiYAaK1XKK8/VUr5ZHAMOAKpdQL
QBhgV0oVaq1fP9AmxOtNW8s3kOXqEAu6FtFV4MzdWgt7PweJjwKfqFmW59LYdu3cOBXs1hFVTIdJ
RlZexq/20TaBvANhYguJolc84EpCWnTtfHu2Zh+eRVWvXv69txD5h8pQoj6CYrBv0hGkoUQzVdtkuTV
QHelVGdMcnw1cG25NgHWpBP3JjSjgy5lGntejX2vgDRWAp5GgCDZnc/GLMpUCKPQYAYCiwZHi5vPNDq1bYnb0D3RyZmUnADAi1HU5U6PiZeaBrt5lUMjo0jYFFmzqS0M4NotvGXOw/DzMYwt3dPe
dT/frN6W/E31SXKWYwS5MAfysyCwTePECI4R1gEmEW83oGxbS0ySbaXw6+umrKJ9hdZtFisMuNY9c
QnhCQKj8LIVEECh1CQLIZqlGpNkrXWpUupOYAHmUouAYAFgBT7QWicrpZ4A1mitZ1ZVP2VG94RCdnOAhhGPANHTyArJvaFbXhR4YuT4NNHqxIJV7wiwpdYMi6Y4Hn9v1smT/I04UOx0m9OCgO9hmRcNy9j8XPM
sfuYLeb5Yi9fNFa8/pPu2kf5s8lfSNqP1JbWmxmeNfm2IOrYddCOPtv4Btctt03CHqcD1v/BxOfB2sVf5yuZ

RZZeypPkrU2o6POP7agGPDyqTk2V6VFpuvDiDvN8+jeYPU1SXLiFXW7VlXsNhNjVa+1Ie1fbhZFmfQ
yJFzc+PcTojUJMn3kI1UORTKSLIRohmpVk6y1nqe17qG17qq1ftqx7TFHgozWeqvWepTWur/WeoDWem
El13hca/1Sw4bvJvMfhrfPAlsJK/dmse5ANvcNC8T7lV6w4eOaz9ca3p0AMydB0Ymaj1/5FvhHwNBbT9/X
51KTyO1fXvX5WXvA19F8pKq65KUvwit94NW+5ufz62uOq6L0ZPOPB2dbNKs3tE2Ewxvrfq2qLPyrie/o
9oa7ZlWSvwbvQOh+buPfS4jWJsjMP4kiW8othBDNUiP1KfNwB36FYzsoWvsxL8zfTmSQD5ecmAVFub
B3ac3np62D9M0msf3ocijMrf749K3QcYQZOa6o+3ngHWBKLqqSuQc6jwFlqbpl3M75ENUbLn4duk4wsd
nr+BfX4Q3mt2tbtNiBpk65rteqTGmx+UfIicPmHxjpW8/8mlWxlcLWOdDzgvJLSwshGoZj1b1IlUOJzXO/e
BRCtFySJNeV3QbHdgJwfP4zJB88xjMTwvDa8KHZX5uWZ1u+Bos3XPxPMyHvo8uqTpTtdji+F9p0qXy/
TwD0mAjb5pjEriJbCWQfgKhepl1ZViUjyYW5JpHtNQkGXQd9LjMJf216MJ88Bpu+MO3otn1r6qbDXJa
ZjR0AxScqv29dpSwxq/id9zRYvOA/k6tPlIvz4eCq07dn7ITcw9Xfa99SKMgyI/VCiIbnKLeIUjkykiyEaJYk
Sa6r4/ugtJDFlhG0tacze9Q+zsv8yNQWD7oBju2qfmRYa0ieDd3OhkHXw9SZkLoa1rxf+fG5h6C0ECKqmf
jW7yrIz4Tv/nD6iO3x/aBtZuJcm66Vl1sc+M0c09kxz9JZLlGbhP/Hx+Hr38HXt8Cen6DTqPJ11rGDzO/U1T
VfqybJ35jOGUNvhRvnmUR5XjUrnf/yCrx/bvlFVApz4P1zYOFfqr/Xlq/BJxi6nXPmcQshThfQBo0ySbKthj
aWQgjhBpIk11WGqYX9QE/mZPRAem9/E9Z/ZBLeXpMBDUc2VX1+6hrITS0boex9EUQnVF2m4RyBra
47RI/zYcyDsP5DmHNX+b7JzvMjupqfrJTT+xbvW2pWx+swzDyP6gVefrVLkvf+DN3OhbvXmZ8rPii/P6q
Xqafe90vN16pOaRFsnwu9J5sJhW26QtLNsH9F5aPCWsOWr8zjJc+Wbf/tLZMoZ+6u+l62EjMq3utC8PY7
s7iFEJWzemHzb+OoSZZyCyFE8yNJch3Z07cB0DVhMIHn/RXyjpha39F/LKvFTdtQdkL2AVjxukm8wIy
GWn1MratT/GgzmltafPoNM12S3KooBRP+AuP+bGp2v73n9POdI8lFuaZEwtXeZWaFPGftrdXLMeHO5
XXYSmDpS+XPPb7fvL7u55Zd38u3/LUtFogfZe7h6tBaWPdh+W1amxUEM3ac/hr3LIainPLlD30uBbTp7l
HRkc3mHwgxibD7B1N2UXAcfn3D7M8+UP74ojxY8Bf47j746mZT1iGlFkI0Kpt/JJEqRxYTEUI0S5Ik11
Hugc2k6kgGdu9oJrj1vcKM4oa2h6BoCIkrPwK75Dnz1f4XM8xoaPI3ZuTVuSAIQPxZUJJvJvRVlJVieiEH
12KRknEPw4i7zMj2McdIadYeU6IQ0KYs0XatDy7INiPfFfssxw40XSmco9I7F8BPT8Kqd8qOcY4Ox59Ft
eLHmFX/ju8r27bwMTPqfWRL2baUJTD/IfOeVZT8tVltsMu4sm1RPSCmb+WTFpO/AWWFaZ9AQKQZ
Tf71TfOPhMQrTcLsWhaTstj0RE7+xoxOxw40f75CiEZjD4yWmmQhRLMlSXId2dK3ssvenmGd25gR3Cv
eh7F/KjsgdkBZklxaDNu+g/B42P4dvHcOnEg7fYQyfjSgTh9tBTMSHNHFjMjWxvA7zO+t35Sd36aLidVZ
suFal3zgV9D20xPd2IFQnFdWluBMRJO/KSvX2LfMJN9RvaqPyVnr7Hx9J46Utaz72ZEQa11WFrFzvpl05
1RSCNvnmdIUq3f5a/e5BA7+Bjkui0BqbZLqLmMhrCOMutfUSy//ByRMKRvFdx1Ndr4n926EP+2GW5ec
PiouhGhQOjCaSCRJFkI0T5Ik14WtlJC8vaT7daZtaBW1qrEDzUhtQbYZnSzKgQteND9HNpnFNXpOLH9
OQIQZEd1XSV1y1p6qO1tUJjTO1BYnzy473zmCHNbRjK66jiTvXWZiihty+usAk/CXFMCO701t8bGdcH
SrSUT3LjMJfk0JfFQvM5rrHHneOgfQpovGtm/NiPWeRXBwpdlWkm8WT3HatdB0yKis/KHPZY5rupRc
HN5gRq2d+4b8zrSbshXD2Ich3NF9wzVJztpjYnQd4RdCNCoVFG1awJXKxD0hRPMjSXId2DJT8KYEa0
zvqg9y1iUf3mg6JDhLBIbdaia1Tfp7+VXznDqfZepmS4vKttltJtmrrh65Mn0ug/QtppQhJ7VsBNnqbRJE15H
kfcugw9DTJ6hF9jD9l9PWmyS15CRc+KKpv97ytYkrN7XmUgswo9jxo829nKO80Qlw0asmKV3yHCx+1r
Sou+RNk9Amf23O1Rp+edm0les85vRrt+kKbfuVHQ+OFntepqUdmDZ5U16Hic9BTEJZi7pyI8kpLXPpbC
FaMBUcjb8qRtdmUSUhhGhikiTXwaGdpmY4qsuAqg9q5xiBPbgSdswr68YA0Pdy04e4MvGjTau31DVl23
IOmtHPuiZvCRcDyrRA0/bySXZE17KR5PwsM8GtskTXYjXJZ9p6U2IREAkJl5hjk78xCS9UnrhWpvNZp
p3d/uWmxKPPpSZBHnGXeZ8OrYExD5jJgwlTYOdCM5lu53wTw5g/nV5q4dTnUtNi7ug2U2ucPBu6jDcj
9E49zofht5vHAW3MPwCy95ftdx1xF0I0CUuw6ZXsVXCshiOFEKLpSZJcB0dTNgDQKzGp6oMC25iyhp
Vvm0lite2Q0GkkoMqST6hdZ4vKhMSaFfqco6uuSXabrmbUVGvHJDxdVjNcUexAOLzJTNpLmGK6XvS
51CSUK/9lRnwje9QupnhHMv39Q+a3830ZdrsZbQ/rCAOuLdtXWmAS5MXPQHhn6H911dd2XuvN4fB8
vJkkWN37rpQZTXaOJBefNKv41aWsRQhxxryC2wLgU5Dh5kiEEOJ0Xu4OoCWxp2/jsIqmXVRk9QfGDj
Q1sv4R0Hls7S7uHw7t+pk633EPm23OFe/qUwbQ9zI4sMI8jnBJ/iK6mtKJ+Q+bRL7vFdBheNWvY+Vb5v

Gpvs4Xw9z7TTlHn8vKLxxSncjuZoWt9C2mLVtkd7PdLwSu+9p08HCOFHccAUFtYeFfzUTHS96qehQZI
KIzXPURZB80z739IHFq9fGEdTQt7KDsfZaRZCGalCXYLE3tUyQjyUKI5kdGkiuw2TWvLdrF/C1H0C6
LbtjsmvCTe8gO6lbzRZyT3irrxlCd+LMgdZWZKAdmJNkn6NTyrXXS21Fy4R9evuzAOVq68m2zUt9l71Q
98c75OoJiHCPdmJHyLo7Ev6oR6Mo465LBdKRw1X6wqRV2sljMMSfSTOKaeGXN1+99EYz4vflJuqmsx
KUqYR3LRpJde0kLIZqO47PNV5JkIUQzJElyBW//vIeXf9jJ7R+tZcoby1m0LZ1d6Sf4KTmVTjqt+kl7Tp1
Gmd/9rqrbzbuMMzXIO+aZ51l7zChpbUdrXQXHmKWv2/Uvvz06wUy+6z/NjNBarFVfo003U1KROLX8c
f2uNhPjuoyvW0zdzwOLtxnlrokzMR7/iCnzaGjhnUznkYJsl1UJpdxCiCYVEIENhX9RprsjEUKI00i5hYuN
B7N55YedTEpsx7ieUbz64y5u/o+ZSNdVHeJcXxvRXQfUfKEOQ+H+HeCot6u1rhMgsif8/IKZJJe5x6x8V1
9T/wNUWO41JBbu22piqyn5tljg97+d3o2j35Vmwl5ILRY4KXfeVeYfArV5X+IGwx+31/0etRXW0fzO3m9
qtINiKu86IoRoPBYrxwkloFiSZCFE8yNJssPJolLunbWemBA/nrk0kdAAby4eEMuynccoLLXR7tBRWAlh
nfrV7oJ1TZDBjNaOewi+vAk2f2kSuIqlCXXhG1T59roknoFtTt+mVP2SV6Xq9r40VoIMLknyAelsITyOU
moi8A/ACrynta5kGUtQSl0OfAkM0VqvcdneEdgKPK61fqkxYz2uwggoyWrMWwghRL1Ikuzw9LxtHMjK
Z9atIwgNMHXEvl5Wzklw1ANvWwS+IRDVs3EDSbgUol6EBY+AvVSSt8bi7JV8fL8Zse9xnnvjEaKBKK
WswBvAuUAqsFopNUdrvbXCccHAvcDKSi7zMvB9Y8cKcNwSTnSpJMlCiOZHapKBUpud/60/xBWD4x
jaOeL0Aw5vMivDDb/D9PFtTBaL6W6R75jIInWyjcM/HHyCIT0ZTh6Vf4wITzIU2K21TtFaFwOzgCmV
HPck8DxQ6LpRKXUJsBdIbuQ4AcixhBMsSbIQohmSJBnYkpbLyWIbY3tEV37AkufANxSG/75pAup9sV
mmGqTjQmNRypRcpCwxz+V9Fp6jPXDQ5XmqY9spSqlBQAet9dwK24OAh4D/q+kmSqlblVJrlFJrMjLq
3+c40yuasNJjZvEgIYRoRiRJBn5LMZNGhnWpZBQ5bT3smAsj7gT/sKYJyGKBya/A8DshMKpp7tkahXc
ybeZARpJFq6GUsmDKKe6vZPfjwCta6xozVq31O1rrJK11UlRU/T+ndvsmYMUOh9bW+xpCCNEYpCY
ZkyR3iw4iMsj39J1LnjMrwjmXNG4qHYaaH9F4nJP3QMpahCc5BHRweR7n2OYUDPQFlijT4aYtMEcpd
TEwDLhCKfUCEAbYlVKFWuvXGyvYvX59zIODq8p6sAshRDPQ6pPkUpud1XuzuHRQ+9N3Ht1ulkae8
Cj4hTZ9cKJxOZPk4FjwCXBvLEI0nNVAd6VUZ0xyfDVwjXOn1joHOLVsqFJqCfCAo7vFWS7bHwfyGj
NBBij1CWG/tROdDv7WmLcRQog6a/XlFs565OFdKml15lzWuaYljkXL5OxwIfXIwoNorUuBu4AFwDbg
c611slLqCcdocbPi42Uh2doLDq4Gu93d4QghxCmtfiR5pbMeuXMlSXLaetMFwZlMCc/iHEmWUgvhYbT
W84B5FbY9VsWx46rY/niDB1YJH6uFTZZeXFi4ADK2l1+iXggh3KjVjyT/lpJJ16hAooIrqUdO2wDtBtRv
WWjR/IXHg5ffma1qKIQ4Iz5eFjbg6D8vJRdCiGakVSfJpTY7q/cdr7zUoqQQjm6F2IFNH5hoGn4hcNdqG
DzD3ZEI0Wr5WC3stUWbTj4HV7k7HCGEOKVVl1skp+WSV1RaeZKcnmxWvJMk2bO5drgQQjQ5by9F
sU1Dh2FwQEaShRDNR6seSa6+P/I681uSZCGEaDSBPl6cLLahOwyD43sh76i7QxJCCKCVJ8lr9x+nc2Qg
0cF+p+9M2wABbSA0rsnjEkKI1iI2zJ/iUjvHIx0DEgdXujcgIYRwaNVJ8pZDOSS2DzVth35+EbZ8Vbbz8
AYziiyT9oQQotF0iPAHYJ93d7D6SsmFEKLZaLVJcmZeEWk5hfSNDYJv74HFT8G3f4CC41CcD0e3Sam
FEEI0srhws5DPwVwbxA2BHd+D3ebmqIQQohUnyVvScrFg55IDz8L6D6H/NVCUC7++AelbQNskSRZC
iEYWF25GklOPF8CQmyFrD2yf6+aohBCiFXe3SE7N4kXvfxG9ZxmM+zOMexhKTsJvb4GymoPaDXBrj
EII4ekCfLyIDPLhYFY+jJ0C4Z3hl1eg90VS7iaEcKvWOZJsK2Xwuke43LoMxv/FJMgAYx+G4pPwy8sQ
GA0hse6NUwghWoH24QEcPJ4PFiuMusd0F9q3zN1hCSFaudaXJNtK4ZvbGJb3I3Pa3AxjHyzbF5MAfS8
DW7FM2hNCiCbSIdzflFuAKX0LjDajyUII4UatL0ne+Als+ZLnSq7mUOKdp+8f+xAoC3QY0vSxCSFEK9
QhIoC07AJsdg3efjD8DtjzExze6O7QhBCtWOtLkg+spNivDW/bLjLt3yqK6gl3rIDhlSTQQgghGlxcuD8lNs
2R3EKzYcjN4BsK8x8xLTqFEMINWl+SnLaewwG9AUWf2JDKj4nuDT4BTRqWEEK0Vh0cbeBSs/LNBr
9QmPgM7P8FVv3LjZEJIVqz1pUkF+dDxjaS6UxcuD/hgT7ujkgIIVq9DhGOXsnOumSAAddCj4nw4+Nwb
Jd7AhNCtGqtK0k+shm0nWUnO9A3tpJSCyGEEE0uNswPpTBt4JyUgov+Ad7+8M3tZtK1EEI0odaVJB/e
AMCinPYkxkmSLIQQzYGvl5WYYL+yDhdOwW3hwpfg0Boz6VoIIZpQ60qS09ZT7B/NUcLpW9mkPSG
EEG7RIcLf9EquqO/lENMXfnsbtG76wIQQrVarS5LTAnoB0LeqSXtCCCGaXFx4QNnEPVdKwbDb4GiyL
DAihGhSrSdJLsqDjB1stnemU5sA2gT5ujsiIYQQDh3C/TmSW0hxaSUt3xKngn+EGU0WQogm0nqS5COb

AM1Pue0Z2CHM3dEIIYRwERcRgF3D4ZyC03d6+0PSjbBjHmTtbfrghBCtUutJktM2APDLyTgGdgx3byx
CCCHKiQv3Bzh98p7TkN+BxQqr3m3CqIQQrVkrSpLXU+DflgzCGCRJshBCNCvOBUUOVlaXDBASC
wlTYP2HcCK9CSMTQrRWrSpJ3u/TAz9vC73aBbs7GiGEEC7ahfphtajKO1w4nfUA2Evhk6lQdKLpghNC
tEqtI0kuzIXMXawt7US/9mF4W1vHyxZCiJbCy2qhXWglvZJdxSTA1JlwZAt8fgPYSposPiFE69M6ssWM
7QD8nNOWgR3D3BuLEEKISnWODGRXel71B/U4Hya/AnsWwdw/Nk1gQohWqXUkyblpABywtZFJe0I
I0UwN6BDGjvQT5BfXsAT14Btg+J2w7r+QuadpghNCtDqtI0nOM5M80nUYg2QkWQghmqUBHcKw2T
WbU3NqPnjk3aCssHZmo8clhGidWkeSfOIwpXgREBpNdIifu6MRQohGpZSaqJTaoZTarZR6uJrjLldKaaV
UkuP5uUqptUqpzY7fE5ouapMkA2w4mF3zwSHtoNckWP8RlBQ2alxCiNaplSTJRzhGGAM7SamFEMKz
KaWswBvABUACME0plVDJccHAvcBKl83HgIu01onADcCHjR9xmTZBvnSMCGD9gezanZB0ExRkwdb
/NWpcQojWqVUkyUXHD5FmD5N6ZCFEazAU2K21TtFaFwOzgCmVHPck8DxwahhWa71ea53meJoM+
CulfBs7YFcDO4ax/uDx2h3ceSxEdIU1HzRuUEKIVqlVJMm2nMOk63B6S39kIYTnaw8cdHme6th2ilJqE
NBBaz23mutcDqzTWhdVtlMpdatSao1Sak1GRsaZxnzKwA5hpOcWVb48dUUWi1mu+uBvkJ7cYDEIIQS
0kiTZKz+ddB1OXFiAu0MRQgi3UkpZgJeB+6s5pg9mlPm2qo7RWr+jtU7SWidFRUU1WHwDHN/41brk
YsC1YPWFX14BrRssDiGEqFWSXNMkEKVUR6XUYqXUeqXUJqXUhY7tbp0EAkBJAT4luRzV4cSEN
um3hkII4Q6HgA4uz+Mc25yCgb7AEqXUPmA4MMdl8l4c8A1wvda6yfurJbQLwcfLUrvJewABEabTxeYv
4Ie/SqIshGgwXjUd4DIJ5FzM13arlVJztNZbXQ57FPhca/2WY4LIPCCeskkgaUqpvsACKnzt1+hOHAHgp
G8Uvl7WJr21EEK4wWqgu1KqMyY5vhq4xrlTa50DRDqfK6WWAA9ordcopcKAucDDWuvlTRm0k4+X
hT6xIaw/UMu6ZIAJj0JhDqz4J1i84ezHQKnGC1II0SrUZiS5NpNANBDieBwKpEHzmATiTJJ1UNsmva0
QQriD1roUuAszKLENM4CRrJR6Qil1cQ2n3wV0Ax5TSm1w/EQ3csinGdghnM2Hciix2Wt3glJwwQsweA
b88jKsfq9R4xNCtA61SZJrnAQCPA5MV0qlYkaR767kOlVOAmmsCSAAnDgMgFdIu4a9rhBCNFNa63la
6x5a665a66cd2x7TWs+p5NhxWus1jsdPaa0DtdYDXH6ONnX8AzuGUVhiZ8eRE7U/yWKBSa+YjheLn4a
C7EaLTwjROjTUxL1pwEytdRxwIfChY3IIUPMkkMaaAAKgHUmyb5umrfIQQghRP85FRepUcgEmUT7
vSSg4DstfbfC4hBCtS22S5JomgQDcDHwOoLX+FfDDUfPm7kkgxcfTKNLetGnT5N8YCiGEqIe4cH/ah/m
zeEc9vlls1x8Sr4Tf3oKcin9VCSFE7dUmST41CUQp5YOZBFLxK7sDwNkASqnemCQ5ozlMAinMOkS6
DqOdtH8TQogWQSnFpH7tWLozg+z84rpfYMKjoO2w5JmGD04I0WrUmCTXchLI/cAtSqmNwKfADK2
1phlMArHnHiadcNqF+TXlbYUQQpyBi/rFUmrXLEg+UveTwzvBkFtgwyeQsbPhgxNCtAq1qkmuaRKI1n
qr1nqU1rq/Y6LHQsd2t08CsZw8QroOp32Yf1PeVgghxBno2z6ETm0C+Hbj4fpdYNS9ZjR56/8aNjAhRKv
h8Svu+RVmkEE4kUGykIgQQrQUSikm92vHij3HOJZX6crY1QuOgdiBsPuHhg9OCNEqeHaSXJSHr+0k+
b7RWC3SWF4IIVqSi/rHYtfw/eZ6jiZ3OwdSV5tuF0IIUUeenSTnpQNgD4xxcyBCCCHqqmdMMN2ig/h2
U32T5HNNycWexQ0bmBCiVfDsJNnRI9kSIqvtCSFES6OU4qJ+sazel8WRnMK6XyAuCfzCYPePDR6bE
MLzeXSSbM91LCQSIQuJCCFES3TxgFi0hi/WHKz54IosVug6wSTJWjd8cEIIj+bRSfLJY6kABLWJc3Mk
Qggh6qNzZCBndY/ko5X7KbHZ636B7uea0rsjmxs+OCGER/PoJLkgK5V87Utkm4Zd6loIIUTTmTEynvTc
IuZvqUfP5K5nm9/S5UIIUUcenSSX5hwmXYcRGy6r7QkhREs1vmc0ndoEMHPFvrqfHBwDbfvB7kUNH
pcQwrN5dJKsThzhKOHEymp7QgjRYlksiutHxLN2/3E2pWbX/QLdz4UDv8GJeoxECyFaLY9Okn0KjnK
MCEL9vd0dihBCiDMwNSmOQB9r/UaTB043k/h+/L8Gj0sI4bk8N0nWmsDiDPJ9o1BKFhIRQoiWLMTP
mysGx/HdxsOk59axHVxEFxhxJ2z8BA6ubpwAhRAex3OT5JIC/HQhtoBId0cihBCiAdw8ugt2rXlj8e66n3z
WAxDcDr7/E9jr0SVDCNHqeG6SXJAFgFdgGzcHIoQQoiF0bBPA1KQOfLrqAKnH8+t2sm8QnPsEpK2H
DR83ToBCCI/isUlyaV4mAD4hkiQLIYSnuOfsbiil+OeieowmJ06FuKGw9IWGD0wI4XE8Nkk+mXMMAJ
8gKbcQQghP0S7Un2uHdeTLdansPXaybicrBb0nQ/YByM9qnACFEB7DY5Pk/JyjAPgEy0iyEEJ4kt+P64a
P1cKrP+6s+8kxfc3v9C0NG5QQwuN4bJJclGvKLQLDZLU9IYTwJFHBvswYFc+cjWlsO5xbt5OdSfIRSZ
KFENXz2CS55IRJkoMkSRZCCI9z+5iuBPt68dKCHXU7MTgGAqMgPblxAhNCeAyPTZJt+Vnka19CQ4
LdHYoQQogGFhrgzR3jurFo+1FW76tjfXFMX0jf3DiBCSE8hscmyeRnkU0gEYE+7o5ECCFEI5gxMp7oY

F+e/347WuvanxjTB45uB1tp4wUnhGjxPDZJthRlk0Mw/t5Wd4cihBCiEfj7WLn3nO6s2X+cxTuO1v7Etolg
K4LMerSRE0K0Gh6bJHsXZXPSEixLUgshhAe7MqkD8W0CeGnBztqPJsf0Mb+lw4UQohoemyT7luRQ4
BXq7jCEEEI0Im+rhbsmdGfr4dzajyZH9gSLtyTJQohqeWySHGDLpchbkmQhROujlJqolNqhlNqtlHq4mu
MuV0pppVSSy7Y/O87boZQ6v2kiPjNTBsQSF+7Pa4t212402csHonpKGzghRLU8M0nWmkD7CUp9w9w
diRBCNCmllBV4A7gASACmKaUSKjkuGLgXWOmyLQG4GugDTATedFyvWfO2WrhjXFc2HMxm+e7
M2p0U00fawAkhquWZSXJxHt6UYvcLc3ckQgjR1IYCu7XWKVrrYmAWMKWS454EngcKXbZNAWZp
rYu01nuB3Y7rNXtXDI6jbYgf//xpV+1OiOkLJ9JkeWohRJU8MknWjg89S0CEmyMRQogm1x446PI81bHt
FKXUIKCD1npuXc9trny9rNw2tgsr92axam8tEl/n5L0j0i9ZCFE5j0yS87IzALAESpIshBCulFIW4GXg/jO8
zq1KqTVKqTUZGRkNE9wZunpIRyICfZi5Ym/NB7dNNL+l5EIIUQWPTJJPOpJkn6BIN0cihBBN7hDQw
eV5nGObUzDQF1iilNoHDAfmOCbv1XTuKVrrd7TWSVrrpKioqAYMv/78faxcMTiOhcnpZJwoqv7goGgI
jIaUxWC3N02AQogWxSOT5IIckyT7h0qSLIRodVYD3ZVSnZVSPpiJeHOcO7XWOVrrSK11vNY6HvgN
uFhrvcZx3NVKKV+lVGegO7Cq6V9C/V01pAOlds2Xa1NrPnjoLbBrIXx7N9htjR+cEKJF8cgkufiEmd0cIE
myEKKV0VqXAncBC4BtwOda62Sl1BNKqYtrODcZ+BzYCswH7tRat6jssWtUEMM6RzBr9QHs9hrawY
35E4x9CNZ/BN/cJstUCyHK8XJ3AI2hNM8kycHh0W6ORAghmp7Weh4wr8K2x6o4dlyF508DTzdacE3g
mmEduXfWBn5NyWRUt2oGS5SC8Y+Alx8s+j8IjYNzHm+yOIUQzZtHjiTb87PI036EhwS5OxQhhBBN7
Pw+bQkL8OaTVQdqd8JZf4T+18CK1yFzT+MGJ4RoMTwySbYUHiebIEL8PHKgXAghRDX8vK1cPiiOh
clHOJZXwwQ+p3MeNyPK8//cqLEJIVoOz0ySi7LJU8EopdwdihBCCDeYNrQjpXbNv5fXoh0cQHAMjHs
Ydi2AnQsaNzghRIvgkUmyb3E2J60h7g5DCCGEm3SLDuLCxHbMXL6PrJPFtTtp6K0Q2QPmPwwlhTUf
L4TwaJ6ZJJfmUuQd6u4whBBCuNEfzu5OfomNd5el1O4ELx+44HnISoEPL4WTmY0boBCiWfPIJDnQlk
uRT5i7wxBCCOFG3WOCmdwvlv+s2EdmbWuTu06Ay9+HQ2vhvQmQ8jMsexneGg3/uUgWHhGiFfG8J
NluJ1jnYfeVkWQhhGjt7j27GwUlNt6p7WgyQOIVMGMuFJ+E/15s2sOVFsLepbD928YLtiEUn4SiE+6OQ
giP4HFJsi7KwYod7R/h7lCEEEK4WbfoYC7uH8t/V+wnt7Ck9id2GAK3LoGL/wn3boQ7V5p65cXP1rw6X
/ZB2D4PdA2LmTSGL26E98+TFQSFaAAelyQX5poaMkuAJMlCCCHg2mGdKCixsWL3sbqdGBoHg66H
8HiwWE33i4xtkPxN5ceXFMLPL8DrQ2DWNPj+oaYtz8jPgt0/wtGtsPnLpruvEB7K45Lk3KyjAHgFtXFzJ
EIIIZqDQR3DCPbz4uedGWd2oYRLIToBljxXfqRWa0ieDW8MhcVPQ4/zYMjvYNW/YM5dYCuBtA2w4
p9wcPXp17XVYYS7OjvmgbZBYDT8/FzDLLO9/1dZrlu0Wh632kZBjhkp8A2WJFkIIQR4WS2M7hbJzzs
y0FrXv4e+xQLj/gyfXwcLHoEOw8wCJMv/AQd/Mwn09f+DLuNM4hwYBUuehS1fQ2mBuYZvKNyyCC
K7m+er34eFf4Vpn5jznDZ/CUW5kHRT7ePb9i2EdoQLnoNZ18DGT2HQdfV7rQAHVsK/J8KFL8HQW+p
/HSFaKI8bSS7MNSMFAWFRbo5ECCFEczG2RxRpOYXsPpp3ZhfqfZFJZle+DV/eaMoqslLgon/AbcvKEl
2lTHnGlDeg31S47F245SewesOn06AwBzZ9DnPvNwn0/+6GIkdsqWvg61th3oOQd7R2cRXmwp6fTHw9L
4TYQab0o7SWPaIrs/lz83vL1/W/hhAtmMclycV5piY5UJJkIYQQDmN6mL8TzrjkQim4bjY8uBfu+BVu+A
7uWQeDZ4C1ki9nB043k//6XQntB8NVH8Lxvaad3De3Q/xouO4byDkIi54wnSm+uhkC2oC9BNZ/WLu4di
0EW7FJkpWC8X+BnAMw+3bYuRCK8+v2Om0lpvba6gMHfoXcw3U7XwgP4HFJsv1kFgCh4ZIkCyGE
MGLD/OkRE3TmSTKYJDQgAmISoPNZ4Btc+3M7jYQLX4TDGyF2AEz71Iw+D73V1DB/fCVkH4Ar/w
vxZ8GambXrVLFtDgTFmBIQgG5nQ9LNpsvGJ1Phhc6wcVbt40z5GfIzYfwjgDbXr46tBN4cAb+8Uvt7CN
HMeVySrPOzyNUBhAb6uzsUIYQQzcjYHlGsTMkiv9jNE9GSboIbvjUj0s4E++zHIKwjHFgBY/4EnUbAk
JvNaPDuH6u/XnE+7PoBek02ddNgEvnJL8ND+8xIdfskmP172PZd7WLc/AX4hcHwO02tdVUdPZx2/WC6
aix+1pSfOGXugR8eg9y02t1XiGbE45Jka+FxclQwXlaPe2lCCCHOwJgeURTb7KxMyXJ3KNB5DPiFlD33
DTKjx2fdD2MeNNt6TTajw2s+qPo6uYdhxWtQkm9KLSry9jOrCF7zGbQfZOqoU5ZUH1txPmz/DhKmm
KW6Ey6BA79VX3Kx8RPwjzA11/MfMdtOZsJHl5mJjW8Mh/UfnXnv6CObyyfhQjQij8skrcU5nLTU4asvI
YQQrcKQ+Aj8vC0NU3LRGGIHmhFlZ22z1dv0ad65wJRgOGltul+8MQxe7mU6aMQkmvrmqvgGwTWf
Q5tu8Ok1cHhT1cfunA/FeWblQYA+l1Cu5CJlCWz4tOz4/CzYMR/6T4OxD8LO702njc+mm8T68vehbV/4

353w8VQ4Wcd+1a73mTkJPrvOPQu1iFbH41rA+ZbkcMIaUvOBQgghWhU/bysjurRhyY6jaJ1Q/1ZwTWn
wDFj2d5MY9rsS2g2ApS9CymJo2w/OfRK6jDVJsqWGca+ACFN68e4E0yLulsUQFGUWPFn1L9PLObit
WX47uB10GmXOi+ppSi7WfWhKP3YtNNv9QqDXJJOw20tgwDVmVcL1H8HnN5iezZe/b5LtPpfB6ndN
u7u3z4KpM6HjsLq9Fz8/b7qCFObAvl9MPbgQjcjjRpK9bQWUege6OwwhhBDN0IRe0ezLzGdPxkl3h1I7o
XEw6e9QWmh6M8+80LSIu+BFs2z2qHugXf+aE2Sn4LZw9cdwMgM+vx5OpMOnV8H8h03i/esbkLbOJL
wWa9l5CZdA+mbTO/ncJ02CPuceyMuADR9D20QzWuzlY/oqA4x9uGw02mKBYbfBzQvNCPnMC2HWt
WU/B1ZWH/ex3bD6PTNa7R9hWvAJ0cg8byTZXkCpRSbtCSGEON3ZvWP46/+SWbQtnW7RQe4Op3aS
bjI/x/fBwVWm60VIu/pfL3ag6d/81c3wj36me8aFL5lVAgEKs82kPVfDbwe/UJP0BkZC9/PgX2Pgkyvh8Aa
Y+FzZsV3Gwp92m5Hr0+49AG5bCt8/aEauLVbIS4eDK+GOFRAUbY6zlZpWeRFdzDE/PGYWbjn3CQiJ
hWUvm/cjPL7+74MQNfC4kWQfirBZ/dwdhhBCiGYoNsyfPrEh/Lgt3d2h1F14vCm5OJME2SnxCtNLOTw
eblpgVtRTyvz4h5vfrvxCTaIcGGmeR/eCc/5mRp0tXpA4tfzxlSXITv5hcNk7cNcq+P2vMGOu6Q89+w5T+l
GYYyb8vZ4EL3YzI8075sJZfzRJ9JDfgbLAqnfP/H0QohoelyT76SJsXpIkCyGEqNw5vWNYu/84mXlF7g7
FvcY+CHeuhLjB9Tt/2B1mdb8B15Qlz/UR3RvOf9rUOy/6P3j/fNi/HMY+BN3OMfXH4Z1h+O/N8SGxpvP
PGug/LVikUohF4VrmF1vhRhM0rwN2RCCGEaKbO6R3DPxbtYvGODK4YHOfucFoui8UshtIQkm6G3Ytg
+avgGwrTvypb4ttWCtpu6p2dht8ByV/Dwr+YUhGrt9lefNJ01Ijs1jBxiVbNs0aSS4uwoLFLuYUQQogq9G0
fQkyILz9ubYElF55KKVMnPfRWuHlBWYIMpiWea4IMEDfELHSydqZZ4jv7oJnY99pAeGOoqVcW4gx5
VpJcYtam114ycU8IIUTllFKc0zuGpbsyKCypxZLPomkERJglu6N713ysUjDxGdNi7vBGeDUR5t5vVi3UN
rNioBBnqFZJslJqolJqh1Jqt1Lq4Ur2d1RKLVZKrVdKbVJKXeiy78+O83Yopc5vyOAr0sWmpY/2lnILIYQ
QVTsnIYb8Yhu/pmS6OxRxJhKvgFt+MhMHp30GN/8AHUfCps9lwRFxxmpMkpVSVuAN4AIgAZimlEq
ocNijwOda64HA1cCbjnMTHM/7ABOBNx3XaxSlRY6RZG8ZSRZCCFG1EV3aEOBjlZILTxDdGy5/F3p
ONCPM/abCsZ1mhLm2Dq01fZ/t8s2CKFObkeShwG6tdYrWuhiYBUypcIwGnMvchQJpjsdTgFla6yKt9V5
gt+N6jaK00DHLVUaShRBCVMPP28rYHlEs3JqO3S4jjh4l4RKweNet5GLZy7DuP3B0a6OFJVqe2iTJ7Y
GDLs9THdtcPQ5MV0qlAvOAu+twboMpKXSsoCRJshBCiBpM7NuWjBNFrDtw3N2hiIYUEAHdz3Usl12
LkeGCbNj1g3m8/9dGDU20LA01cW8aMFNrHQdcCHyolKr1tZVStyql1iil1mRkZNQ7CJuj3EL5yLLUQoj
WqxbzSG5XSm1WSm1QSv3iLKFTSnkrpf7j2LdNKfXnpo++6UzoFY2P1cL8LUfcHYpoaP2uhLwjsG9Zz
cdu/w5sRWD1hQOSJIsytUlkDwEdXJ7HOba5uhn4HEBr/SvgB0TW8ly01u9orZO01klRUVG1j74CZ7mF8
pGaZCFE61TLeSSfaK0TtdYDgBeAlx3bpwK+WutEYDBwm1IqvkkCd4NgP29Gd4/k+y1H0DLJy7P0mA
g+wWYCX002f2kWK+l9kUmS5b8F4VCbJHk10F0p1Vkp5YOZiDenwjEHgLMBlFK9MUlyhuO4q5VSvk
qpzkB3YFVDBV+RzdHdwuIrI8lCiFarxnkkWutcl6eBmHklOH4HKqW8AH+gGHA91uNM7NOWQ9kFb
Dnk0S+z9fH2NxP4NnxieilXJe8o7P0Z+l4OnUbAicPSY1mcUmOSrLUuBe4CFgDbMF0skpVSTyilLnYcdj
9wi1JqI/ApMEMbyZgR5q3AfOBOrXWjTR21FzmSZCm3EEK0XrWaC6KUulMptQczknyPY/OXwEngM
Gbw4yWtdVZlN2moMjl3OzchBqtFMT/5sLtDEQ3t/GfMstbf3gu/vVX5McnfmNX8EqdCxxFmW1UlF0V5
MsrcytSqblhrPU9r3UNr3VVr/bRj22Na6zmOx1u11qO01v211gO01gtdzn3acV5PrfX3jfMyDHuxqUn28pOJ
e0IIUR2t9Rta667AQ5g2nmBGoW1ALNAZuF8p1aWK8xukTM7dwgN9GN4lQkouPJG3P1z9MfSaDPMfh
vmPQNGJ8sds/hJi+kJ0L4jqDX5hlSfJqWvgxa6w/qMmCV00Dx614p52JskykiyEaL1qNRfExSzgEsfja4D5
WusSrfVRYDmQ1BhBNicT+7YjJeMku4/muTsU0dC8fGHqfyDpJvjtDfhnkkl0k2ebpDl1lSm1ALBYoOPw
0ztc5B2Fz66D0kLYMa/JX4JwH49Lkku1BW8fX3eHIoQQ7lLjPBKlVHeXp5OAXY7HB4AJjmMCgeHA9
kaP2M3OT4hBKZizMa3mg0XLY/WCya/A7xZBaHv4353wxQ2w+j3oMg4GXV92bMfhkLkL8hwlRLYS+
GIGFByH+LNg7zKwlbrjVQg38HJ3AA2qpIB8fPHxbrRF/YQQolnTWpcqpZzzSKzAB855JMAaR5ncXUq
pc4AS4Dhwg+P0N4B/K6WSAQX8W2u9qelfRdOKDvFjfM9oZq0+yN0TuuPj5VHjR8IpLglu/hH2LgHfU
GibCF4+5Y/pONL8PvibqWf+/kHYvxwufceMSn9xg1mdr+OwJg9fND0PS5LzKcRXPuCEEK2a1noeZmE
n122PuTy+t4rz8jBt4Fqd64Z34saZq1mQfISL+se6OxzRWCwW6Dqh6v2xA8DLD1b+y5Rj5ByAUX+A/ld

BfhagIGWxJMmthEdlk6oknwLtg4/Vo16WEEKIRja2RxQdIwL48Lf97g5FuJOXL7QfbBYh8Q2GGXPh3P
8z+wIiTBK9Z7FbQxRNx6OySeUst5CRZCGEEHVgsSimD+/Iqr1ZbD8iPZNbtfOfgUv/BbcthfjR5fd1GQ+
pq6FQ/htpDTwqm1S2Aim3EEIIUS9TB3fA18vCh7/KaHKrFjsA+l9tJvxV1HU8aJupUxYez6OySWtpAQX
aB18vmbgnhBCibsIDfbiofyzfrD9EbmGJu8MRzVGHYeDlLyUXrYTHJcn5+OIrI8lCCCHqYcbIePKLbby3
NMXdoYjmyMsXOo2ElCXujkQ0AY/KJq3OcguZuCeEEKIe+rYP5eL+sfxraQqpx/PdHY5ojrqOh2M7IPtg
zceKFs2jskkvWyGF+GKxKHeHIoQQooV6+IJeKAXPfe/x66iI+ug1GSxe8MvLDXO9b26Hz6Y3zLVEg/K
sJNleSJHFz91hCCGEaMFiw/y5bUxXvtt0mNX7stwdjmhuIjpD0s2wdiYc3XZm10rfChs/hW3fwYn0BglPN
ByPSpK9bQWUKFmSWgghxJm5fWxX2oX68X/fJlNcand3OKK5Gfew6aO88K9ndp1fXgarD6Bh+7cNEp
poOJ6TJNvteOtiSmQkWQghxBny97Hy18kJbDmUyx8/34DNrt0dkmhOAiJgzJ9g9w+w6wfYOgdmToYPL
3OszFcLWSmw5SsYeitE9oCt/2uY2LL2gpb/XhuC5yTJpQUAFEuSLIQQogFcmNiORy7sxXebDvPo7C1o
STyEq6G3Qng8fHwFfH4dHN9nVur7YCJkH6j5/F9eBYs3jLwbEqbAvl8gL6P+8djtsOAv8NoASP66/tcRp3
hOklxsZiHbrJIkCyGEaBi3junKXeO78emqA7y4YIe7wxHNiZcvXPSamch31cdw70aY/jWcOALvnQs/vwi
LnoAf/mZGjV3lpsGGT2DgdAhua5JkbYft39UvlpIC+HIG/Po6oKRFXQOpZDmZFqrEJMkl1gA3ByKEE
MKT3H9eDzJPFvPmkj2M7BrJ6O6R7g5JNBddxpofp85nwU3z4dOrYPFToCyAMgnxdd9A274mif78epM
Uj7rHnBfTFyK6mpKLpBvrFoPdBh9PNSPR5z0Ne5fC/l8b7CW2Zp4zklxiyi1kJFkIIURDUkrxt4sS6BIVyI
NfbpTV+ET1YhLgng3w6FF4LAt+/5tpGTfzQljzAfxrrOlqMfXfplwDQCkzmrx3KZzMrNv9Nnxsyjwu+ge
MvAs6jYDMXXDyWNkxKT/D4mca6hW2Gh6UJJ8EwG71d3MgQgghPI2ft5W/T+3PkdxCnvpuq7vDEc2
dxWrKMZSCqB5mdDmgDXx3H3j7we9+NEmxq4QpoG3w6z+h6ETt7lOYa0o6OgyDQdebbR1HmN8Hfis
7bslz8PPzkHv4zF9bK+JBSbJjJNlLkmQhhBANb2DHcG4f25XP16SyaJv0tBV1EN4JbpwP5zwOtyw2o80V
tesPHUfCL6/Ai93gs+tq7p287CU4mQETnzUJOUDsQLD6wgFHyUXu4bLHe39usJfUGnhOkuyYuKe9JUk
WQgjROO49pzu924Vw76wNbErNdnc4oiUJjoHR95n2cZVRCmbMNcn0oBtg53wz+luVrBT47S3ofw20H
1y23csX2g8qG0neNgfQ4OUPexY32MtpDTwnSXZM3JNyCyGEEI3F18vKv2cMISzAmxs+WMWu9Fp+
LS5EbVgspqb4whcg8UqzGl9Bdtn+9R+bUebnOsGbI0wLubMfO/06HUfA4Q1mADH5G4hOgJ4XmK4X0
sqw1jwoSTblFtpbulsIIYRoPG1D/fjo5mF4WS1Mf38lKRl57g5JeKJht5oBwPUfmed5R+H7hyAkFvpdBYN
vhGmfQEi708/tOALspbB9rhlR7nMpdB0PeUfOfCntVsSDkmQzcU9JuYUQQohGFh8ZyEc3D6O41M6UN
5bzw1apURYNzFmjvOod0+Zt0RNQWghX/NuMNF/wHHQZV/m5HYYAChb9H6Ah4ZKyY6WHcq15UJ
JsRpLxkZFkIYQQja9n22Dm3DWa+DaB3PLfNby4YLssXy0a1rDbIHu/qU1e/xEMvx3adK35PP9wU2KRc
9D8juoBYR1NL+YUqUuuLQ9MkmUkWQghRNPoEBHAF7eP4OohHXhj8R6mv7eSoycK3R2W8BS9Jk
NIe5MkB7SBMX+q/bkdh5vffS4t29Z1POxbDqXFDRunh/KcJLn4JMXaC29vH3dHIoQQohXx87by3OX9e
PGKfqw/eJwL//ELK/Ycq/lEIWpi9YIhvzOPz34M/EJrf263c8DqA30vL9vWZbwpT01dVf7YohOQuUcm9V
XgMctS65J8CvDB1+o5eb8QQoiWY2pSB/rFhXHHx2u59r2VXD2kIw+e35PwQBm8EWdg+O+hTTczqlw
XPS+AB3aBf1jZts5nmaWyl/8DtnwNGTvM6nx5jpr6SS/DkJsbLPSWzmMySntxAfn44ePlMS9JCCFEC9O
zbTDf3jWa343uzOdrDjLh70uYszHN3WGJlszbDxIuNu3h6kKp8gkymJHoTqNg10LY/CXYS6DbuXD238
yqfYuegPysBgu9nOJ8eO9cSJ7dONdvBB4zkmwvOkmB9pEkWQghhFsF+nrxl0kJXD44jke+3sy9s9ZjUTC
5X6y7QxMCps0y5RXBbctW6QMz8vzWKPjpSZj8SsPfd80HpsxjeSn0uaThr98IPCaj1CX5FOKLj5RbCCG
EaAZ6tQ3hk1uGM6RTBH/8bCMrdkudsmgGfINMb2XXBBkgujcMvRXW/BsOb2zYexbnw/JXwTsA0tZB
enLDXr+ReExGqYtOko8vvt5Wd4cihBBCAGZS37vXJ9E5MpBbP1zLzOV7+Xz1QeZsTCOvqNTd4QlR3
riHTReN7/4Ih9aBrYH+G13zPpzMgMvfM6sErv+4Ya7rdHhjoyTeHpMkU1Jgyi1kJFkI0coppSYqpXYopX
YrpR6uZP/tSqnNSqkNSqlflFIJLvv6KaV+VUolO47xa9roPU9ogDf/uWkoEYE+PP7tVh78ahP3fLqeK95aw
eGcAneHJ0QZ/zCY+CwcWgvvjofnOsCn0yBtQ/2vWXzSTBTsMh56TYJeF8KmWZW3obOVQGFu3a6v
Ncx9AD6+suGSegfPyShL8inAV2qShRCtmlLKCrwBXAAkANNck2CHT7TWiVrrAcALwMuOc72Aj4Db
tdZ9gHFASROF7tHahvrx4x/HsvKRs/nlofG8c91gUo8XcNmbK9iZfsLd4QlRpt+V8MetZmW/gdPhwK/wzl

j4YgYc31/36612jCKP+7N5PvA6yM+EnfPLH5e5B94aCW+PhtKi2l9/1w+m1nnM/aZlXgPynIxSkmQhhA
AYCuzWWqdorYuBWcAU1wO01q5DNYGAsznqecAmrfVGx3GZWmtbE8TcKvh4WYgJ8SMuPIDz+rTl
s9uGY7NrLntzBXd+so4PftnLjiOSMItmICQW+l4GF74I9240i5jsXAgfX2FGe2vDbodlL8OPj5uezR2Hme1
dJ0BwrFlB0GnPYnh3AuQcMisMbvy0bJ/WsOP7yrtu2O1momF4vEm+G5jHdLdQpabcIkSSZCFE69YeOO
jyPBUYVvEgpdSdwB8BH2CCY3MPQCulFgBRwCyt9QuV3UQpdStwK0DHjh0bLPjWpE9sKF//fiQvLdj
B6n3HmbvpMACT+7Xj4Qt6ERce4OYIhcC0jZvwKMQOhFnXwNqZMPQWs+/gKvhsOmg7BMWYjhmR
Pc0y2Fv/B3t+Miv+XfSPsutZrDBgGvzyCnx0OeQdNfXEkT1g2qfw5Y3wy6swYLoZGd7yFXx1M/ScBNM
+KR/b9m/hyCa45G2wejf4S/eYJNlSWkABvrKYiBBC1ILW+g3gDaXUNcCjwA2YvxNGA0OAfGCRUmq
t1npRJee/A7wDkJSUJMt01VNceACvXj0QgMM5BXy66iDvLN3Dwq3pTB/WiSsGx5EQG+LmKIUAel4I
nUbDkmdNSYat1JRgWH2h29km2c056Fj2ugC8/GDyqzB4xumdNAbPgJ0LzOhwcDvoPAbGPgR+IXDW/
SbxTv4G4kfD3PvBJxh2zIWUJdBlnLmG3QaLnzHJdb8rG+Ule1ySLOUWQohW7hDQweV5nGNbVWYB
bzkepwJLtdbHAJRS84BBwGlJsmh47UL9+eO5Pbh6SAdeXLCD//66jw+W76VX22C6xwSjtcbLorh8cBxn
dY9yd7iitVEKzn8K3hkHS1+Co1tNrfHNC80os5PdDjkHwMsfgmMqv1ZYR7hjeeX7ek6CqN6w7O+w+X
NTn3zLIvj0apj/CNy21IxGL3sZMrbD1JnmeSPwjCTZVoLFXiKLiQghBKwGuiulOmOS46uBa1wPUEp11
1rvcjydBDgfLwAeVEoFAMXAWKARVhUQ1YkN8+eVqwbw18kJzN2UxpyNaWxOzUYpRW5BCbM3p
DEpsR1/mdQbq0WReryA3MISQvy8CPHzpn24PwE+nvHXu2hmYgdCv6thxWvm+aS/l0+QwawMGB5f/3
tYLHDWH+HrWyBjG0x8HmL6wLlPwhc3wOr34Mhm2PARJFwCvafUeMn68oz/i0pMC518fPH1kj7JQoj
WS2tdqpS6C5PwWoEPtNbJSqkngDVa6znAXUqpczCdK45jSi3QWh9XSr2MSbQ1ME9rPdctL0QQEejjDd
SPiuW5E/KlthSU23l2awuuLdzN38+FKzwv29eLqoR2YMaoz7cP8myha0Wqc/RjsmAc9JkLSzY1zjz6XmZ
rl4LZmgROAhCnQcSTMf8g8H/sQjH247st114GHJMn5AGbFPRlJFkK0clrrecC8Ctsec3l8bzXnfoRpAyeaI
T9vK3ef3Z1LBrbnm/WHCA8wI8eh/j7kFZWSnV/Mom1H+WD5Pj5Yvo/4NgGEBfgQHuDD1KQ4zkuIQ
VWsDxWiLkLbwx82gV/Y6bXGDcXqBbf8ZOqdnUmwUnDhC/D1rabbRt/LGufeLjwqSZZyCyGEEK1Bh
4gA7jm7e6X7pgxoz8MX9OKTlQfYe+wk2QXFbDucy20frmVofAR3TuhGTkEJ2w7nkplXRHSwHzEhvo
QF+BDgY8Xfx0r/uDACfT0jRRCNwD+88e/hXcm3IG0T4fe/Nv69HTzj/4BikyTn4ycr7gkhhGj1YsP8eeD8
nqeel9rsfLbmIK/8sJMbPlgFgJdFER7oQ9bJYmz28g1K2oX68cSUvpybUDbxym7XWCwyCi1aD89Ikh01y
YXISLIQQghRkZfVwrXDOjFlQHtW7D5Gh4gAukYF4eNlwWbXZOYVkV1QQkGxjaMnivj7wh3c8t81n
N0rGm+rhU2p2WQXlPDnC3szfVhHKdkQrYKHJMn/396dx1Vd5Y8ffx32VWRzBQV3JbyyuC9gWFkyk
NsUZblMVrY4Wk1De9PUfJvq16mRqRWPPYaOJomeSu5YpL5UYuYOJCiIoiKss9vz/u9QYIagpe78f3
8/HgEZ/zWe773HM5vT33fD7nwnQLd9wlSRZCCCFq5OPuwu0RqTRqUOTspGGxXFzrFEDD1tZfPtgPlyzny
nL9xLk605MWADHz5znpfnbWb+vkCcT2rBidwdTdCMkaSbH26hdnZ4zdv1vtYSoF4jRRCNwD+88e/hXcm3IG0T4fe/Nv69HTzj/4BikyTn4ycr7gkhhGj1YsP8eeD8
kWQgghxDVydXbisfg2jItrbRs1Nps109bs5+3MbNaNaJyNaJaNKSBhzOfrD3Ah2tyAPDzdCU8yJu4dsEkdm5
K28a+dquHENfCUElymXKX+VJCCCFEHak8rcLJSfFoXk8k8vbdHZoFvNAAy9LZZ0sr2JR7nJ7nJ/yT5
Nz7Ay7j55m8vI9TFq2h2h3aNfUju0pwkUzPb8UI4AoMkyZY5yeUu8QjhtiCm6zDpYcil+Zl+7YL
p1+7XlQB/OXWOxTuOsvvD7w7ydmc3bmdl0aOJLy

AkJecreGzWFm5/bzWzN/7MyZJSe4coHJRhssrSCrMsSS2EEELcpLzcXLivewuWPBXHv+6LwsVJkTrvR
2JeX8oDMzYwf+shyiosXzmbzZr/bTvEmI838VP+aTtHLm5UxphugWUkWeYkCyGEEDc3ZydFYudmDI
psyvZDp1i0/QiLfjzChM+38XZmNvd1b0HmjqP8kFeEk4Kdh08xd1xPQvxlyWxR1RVllUqpgUqpbKXUXq
VUag37/6GU2mb9+UkpdbLSvreUUjuUUruUUpNVPc2mlyRZCCGEEBcopYgM8ePPAzuw4ul4pj8YS1M
/D97OzObY6fP8v+EmFj7ZhzOl5Tw4YyOFxec5UHiGGd/m8MXmPHuHL24Alx1JVko5A1OA24A8YJN
SaoHWeueFY7TWEysd/yQQZf29F9Ab6Gzd/S0QB6yso/htSstluoUQQgghLubkpBjQqTEDOjUm99gZmvh
52BYmmTmqKyOmb6DfWys4U1phO8fFWZHcpbm9QhY3gCvJKrsBe7XW+7XWpcBsIPkSx6cA6dbfNe
ABuAHugCuQf/Xh1q60QkaShRBCCHFpYUHeVVbu6xoWwAcPxNCzdSCv/K4Ty5+Oo1t4AM/O/YHvD
560X6DC7q5kTnJz4GCl7Tyge00HKqVaAuHAcgCt9Tql1ArgCKAf2mtd9Vw3sPAwwAtWrT4LfHbnC+
TJFkIIYQQv118+0bEt//1Ocxv748mecp3jE3L4o3BkZw6W4apmvh52DFScT3V9Y179wJztdYVYV
AEqpNkBHIMS6f4f/Wek3lk7TW04BpALbq72DkUIUccCfdyZPjKWoe+vZWxalq3836v2MWNkV25p7mfH6MT1ciVJ8iEgtNJ2iLWsJvcCj1faHgys11o
XAyilvgZ6AmtqOPeqaa0tz0mWZalFPcjLy8PX15ewsDBZxUkDBZxUkAlj6nsLCQvLw8wsPD7R2OEKIedGjSgC
VPxXHo5FmCfdw5da6Mcf/ZwvB/r+OfKVEM6NTY3iGKenYlQ6+bgLZKqXClBuWRHhB9YOUUh0Af
2BdpeKfgTillItSyhXLTXsXTbe4VmUFndxlJFvXg3LlzBAYGSoIsbJRSBAYGyrcLQhhcs4aedA0LICzI
m84hDcl4vBdtG/sw9tMsHvoki/X7C9FaU1Zh5ufCEopKyuwdsqhDlx1J1lqXK6XGAUsAZ2C61nqHUuoo1I
EtrfSFhvheYrbWuPF1iLnAr8COWm/gWm/gWa60X1mkNsNy0B8jTPTLUS9kQZZ2Cm1nqHUuo1I
P/6w/wL3T8gnyceNESRkVZo2vuwuvJEUwNLq59BEGcEVzkrXWi4BF1cperrb9ag3nVQCPXEN8V6S03
Joky0iyMKDCwkISEhIAOHr0KM7OzgQHBwOwceNG3Nzcaj03KyuLtLQ0Jk+efMnX6NWrF2vXrq2zm
CdMmMCcOXM4ePAgTk7ydymEMA5PN2eeuq0dj8W3Zt6WPZpgbO93yz4yipd
3YgPMhbkmUHZogV9yRJFkYWGBjItm3bAHj11Vfx8fHHhmWeese0vLy/HxaXmP+XY2FhiY2Mv+xp1m
SCbzWYyMjIIDQ1l1apV9O/fv86Xdml6Ei2EEPXNw9WZ+7q34N34L7uvz6Va3hsKDO/zeHtzGy+2ZlPkwYed
G8VwICOjUno2AgvN+mzHIkhskpbkizTLcRNYtSoUTz66KN0796796dZ599lo0bN9AgvN+ioqLo1asX2dnZA
KxcuZLExETAkmCPGTOG+Ph4wWrVqUl02cfHx8Z8fHw8w4YNo0OHDtx///1orTGTVqVs1cNEi+0c+fOr
MDOPHj7ddt7qVK1cSERHBuHHjSE9Pt5Xn5+eTnJxMZ6enp5eXRuXNnTCYTDzzzwgzwgK1+c+fOr
TG+vn37kpSURKdOnQC4++ZiYmJMjISIigmnTptnOWbbRt25aCggLAksy3adPG3G
ti2EENfK2Ukxtl8rlj8Tx+t+/3iE9I0/X7Rf2I8h/klzvty
yQo6MJIv69peFO9h5+FSdXrNTswa8kht5UxdnZmVOnzFzzzqWpzz//PF988UV7YvmExD0IhgmnTptnOWbbRt25aCggLAksy3adPG
+zevZsVK1w0b+vRp2rdvz7hx4y56hNnWrVvZsWMHTZs2pXfv3nz33Xd07dqVRx55hNWrVxMeHk5KSkqteaWnp5OSksLgwYN54YUXKCsrw9XVlSeffJK4uDgyMjKoqKiguLiYHTt28Prrr7N27VqCgoI4fvz4Zd/Xli1b2L59u+2xZNOnTycgIICzZ8/StWtXhg4ditlsZuzYsbZ4jx8/jpOTE/fffz+fffYZEyZMIDMzk8jISNuIqhBCXC0Fz1bz3Hc/jZ1n4Md3YlXJrdFPEmVDGdzcZHs/jz5QTvdtM0QGUZ+zcEzp66//5xm5ZFkzs3n2xYsS/6eFurZcWQ3zzzmc2ErdWQWMfYcP+B5LTcPiW3dCkuIDFObkalyZndTbEz+O6jfYbpCdKnNWBGB7e0ugEbr+urTyDMpYI8aQZmdbucsSHOzZsAJ3aCb+sQCz/cxE7sddixB3+YdWbsB7jWNhfbsoXlpn6ATsn8q/qJ7HL3zHTzRi8fA+0uNTlmdwJYxfYXX7wfzGnJl3zRgtIpmg5U3hq02mjOk8zhhh63TaGDSmfzb8wR/UC2kFFcz+vOx9xqahZfAwDCKzSfU0FIOWwmIcNgWlw
ChAnMnDmT0aNHX/b1HI1SyhmYAtbPAe8CAyvtfxf4+ieLSShPTspErpBZWTYOAzrfW/rccnAe8CAyvtfxf4+jqFLIShOTsper
YOpGfrQJ64tQ0vZGzn9a928cYiy8O+fte5GcNiQgjyccfJCV7I2M4Tn21l84ETPH17e3zcDZGqOSRDvPM
Xnm4hj4AT9e1qRnzri7e3t+33l156uJmLy8qs6pjaZmZcPHmSyM
hIAEpKSvD09Xs9FKx1akZtXFxcLmXunV4eXkRk+xD4/yceyhYaG0rhxY5Yv
X87GjRuZuDZGpq9ZtpYfQXh1mp8CmgabGGqDqnTrmBvWqCdUIHIQV+HI/O+HQUJY0ZtFP+nOd/v36ez
BgZy8IfjrDt55Pc1z2UNo18qxwz++Ee/N+i3cz8LofPNvzMrR0akdylGbd3aoKTk+UmwLIKM/9ctoejp87x+

9hQYlr6yw2C9cAQWaXcuCdudkVFRTRv3hyAjz/+uM6v3759e/bv309ubi4An3/+eY3HpaenM336dHJzc8
nNzSUnJ4clS5ZQUlJCQkICU6dOBaCiooKioiJuvfVW5syZQ2FhIYBtukVYWBibN28GYMGCBbWOjBc
VFeHv74+Xlxe7d+9m/fr1APTo0YPVq1eTk5NT5boADz30ECNGjKgyEm8wzYGDlbbzrGVVKKUeV0rt
A94CxlvLfIA/A3+5DnEKcVNSSpFkasbLv+t0UYIM4OrsxMu/68S8x3pxb9dQNuWe4NH/bOHeD9ezv6C
YwuLzjJi+gcnL9/LlD0cY9u913PHeajK25mE2X9WixaIWhsgqLyTJMpIsblbPPvsszz33HFFRUb9p5PdKe
Xp68v777zNw4EBiYmLw9fXFz6/qsqwlJSUsXryYQYMG2cq8vb3p06cPCxcuZNKkSaxYsYsYLIyEhiYmL
YuXMnERERvPDCC8TFxWEymXjqqacAGDt2LKtWrcJkMrFu3boqo8eVDRw4kPLycjp27Ehqaio9evQAI
Dg4mGnTpjFkyBBMJhP33HOP7ZykpCS0tZ23Zykp3uiMHk33NDUh7mvAxExaJNmgqSJGTMyRPPtSKyjaEnzQJ1U4cb6DqFxwtR0RwIwqeV4ix+3qHqLjtPhx6Ng7BdO6AL17BzIzPzPvsIBLIxGwF0jjCYOOT7OhbGoUxB6v8ZSC1g7JZOoP7jxyNLtJXIcU9cQvs1QEHvXfKzBSTxVw3OY9mvXVdUXFAmOvNLqxFb4G7PmgrdwzALJCyw8ooFCxzCvK7eVA7aJVdJZ2K+Jt7sYeHAEvUCaxzaRV2sXkx8fpnXS5lVy89s8MTBtoRkyNyAMKFNvyFfdAk9OMxJFbMbp1qyGRuXQ9LYP1lLchCLMuGD0ugEa1jCvLIu0OVWQDBGWXLfhZcl5Q1YIRdm76MWIBBfVPeI1zjV9+uThRkrR/vvDcOMZYijO4TnlkPrR2QO4tYGH9zJSxEZVB4RZmm5wQHsNBJjBsBVGGGOBKEGiOMAjUTLq2bXJYZ3bw5jH6YlGfTVUrTGeiRZe5qlm+XtGdHUfjA1CAy1bEc7K4v2aDQUsa0F90mtSy4H8KcNqMMDKl0Q55HvebgRnmxJVmNHplM+XBaXu+2fBAgAUeVL9TZvJ8vBVcN3hJ3TQa+dBYc2Zn+u6B/MkT39oNuI/Ocen6pUaTWJxhVKW8zTz/1w7vO7cOa8EmGaX0uDyjJ+4BNJT3rbf5+z7YGmpxWC5H51YpFPOu3H0a5tUbDX/sX9mJHE8EpDHv3wF7H3O0a7cRuR/BTm3qjyNsEfsFdd3nwA5CFeZhp9zrJBtLFFMwUA5eZp+tmIkpuYbJ1WXyePRJOkp5s3M0wqdO7O6Itg1F7Zo/j+yoszWFSMfPL3uJmVT8MRk6T2hI7dT56VZV0AXgT0w3Nn7WfrNf/bOuHl+u+/JXH/AXfPu7yeEkSyjd1Q8ejMHqYMtBxP9DWs3OBLVtgSSjdCdJCuUOJzM82ZSo2ZrvKI3nEZCdE60G7vgRBvm4I3l7GPFyGzvlwSg1NqL1Zu1vIrG1grxUnm1Ng1kNgmBqlJfJlUSeR2r9XCh2dWdpd7m72uSdwSrSj2EmqsEzPAgH12mBJj8GMQaIu1h2wnmrYjbsUGmHVLNk+SLUrrz1U/ycOpPGz8W1pQo5nAxgD4zljkRZzKk6uNncBkONhzJAb3JQmtY9rN7Jp+i23X2cTxt11SM2ZEnzi8u/kuVTeqW7NrT5cdNR7QJDQJ6S8bsLQ5AaW/NpvGLpq1JfMLnTq2aL5Sr8DZZ4V2Bw8XgEqVNaHrvHpH6k7bKkmx9hKyKYo/UuP9iAI+PIQzw95bVs6nQOfJtfyxHlQyOpYPvT8vbbwSS1m/TanYrsrkOEI7QTmZhT0/tZdDJOTaDFLKWRNq/tNpVTGqNOr4wPZW8JbuLUJlcZEbTFUnjqQM5JVlHgtlvF/XEXz0Mwm5sLB+6VmJJIwcA4k8jYkyOkAP0ykhZ2q7jFebiBHkOwWvOoZbiZ2JXWk4OGYAqSM+CfaCNflUu5AHDY7YchQWDKGN0nt2ahV6QhaXFU0k/Yes8eF9ODhk+iZ3pfo5T/DyFjyQdGt+aKIXs2Cy0uxjK3bo8rTsF7mMAnKJf/QuZhLyuSJkjOt9yl3BQLvKQuR0AbeUF4N1HMKdH+V1A/PfPWWb3Tor8B/EvZvGC0xffOdt/zqP1hP01lwcfjEi7NAqjK5jXw/jnY7FMzuO8QyrF2uR9q9Oc/E6vHnrwgyQE5F5hbvfsUf7hPNgsiRF6z+u4xjXgBXtZKPQd30/PBoG84pHjbkdPG8uiAP8Ko4m6G80tZZL+wwDwBrHjebzHZrFm6DzObRXD6A7ZnT0jDwqpHF9k7fi8nNk/IbJTYtOGVx7r66bNm/7GtSRu6ha6XkiPg/d6bW7ca8d5YrTZBXgWPuSbBzfQnmpb3DRPWDsHsPWrxzi9UBGxHMfFxp1mpdXzyBsJzu1pEo8mbYHTOX2Mb6c3NoCvrE26cAWGr8zsuU2SfdGQhy3vLbPV03IJjDrw8ssrXtpsGPq+Td1mK5m8iNQa3+Emgh6FkN3+hRNX8eLpi86Pyqvq07Dq/4jlsL2A0Owp6KFzRpkgzk4gwbbjEhJKf3sUvcWNLyWcbNNbKPzKt+YUbG/RLMSH3BH9ZmYnL2FWOFa2oE6W+0gxF+/eIohIDjzZ3qYLyxT6b3OMw3Luh9kwuP1GcfBGA0pBS/fzZu8ZBSX2bw5pFNEXbaU1Dw/QYmJTvDVCYsT0q9ML/1Bx5ZhgMyqAsnL4/z0h17bp8QA6ahWXuwbRGhEiKNSLNnTcjfe6ZW9YjGyVfPOjKqRFe3+cGvWGDJhBmSO14gFOmw+5NMbpOdS94yO0uwDX0Ds+SYbsUBNHzkKBB3nMl2JQDz9t30b/OVv2yR0Lq+1ce9NDyKdPO3aw2H/xfj7TG8yoy7eGkH+tiQoAxYuTyvKtFvYy+Jaa5SqiLhBBdz/Y+vsQWwHOGhP6NeMzHF3xRvPwF/rPxMJzLo5zOZOGHKt0aASSYUo2FtAn1ON6dR8W7/CHQWbDMLmxNXwAGlIgTSY0uNMwaFqXQgPyjBdfWf8x6A8Wqt/vRZ/NjoyuOBRnb/ZjXT/O+WfD8mNJbTgpf5Be5Fe2TWm4fcb4fUD8l8q4v/5p4JjJgUJO5aGjCM0RvRHjF8bz0V2MNoc1bWdGlT4w6Kd4+8E/WFyWA7snb4VmyKWv3KRTEz/Ab7DIfbBjZbwkSHO1TMz7t88G9/+lEgy9+v/fu4N2PE0F1QNUWxRIeMtIjs+92b55CYU6C3yebgA/CcPgZVKRDSt2XOy+f0tDOrCG9t3J6yPL9CjLY5dZGtSQgmWfUCu71Cx6fuTLvmGJAnTqEh2FPS8GNpG2QKOsIAmdf56NNbPwLU9sSGL9ZBv+4ZMEntzmN/3usHjdCb/HuRMJbUzPV0kn9oZh8EavkRAZ6KAt3Pxb3nzhWR0vUhTgyU0aVH6TK4NuXWFzNtU/X9+gT5xPb7d+rOaLCbA6zw1Z+L3C8lhPWcBaMoJYAMkCkg7/Q8i7JZGKZcdxXOfFVGE8mQ5gH72eEDi+4DV0ZH3+QFSdqFtvDs/R/+Gzrkh2VpWSwJX4v+tgq5aXGtsNM1Bw2ejUYKBV9gLTVHCuWlBzeadQYbj9vVGsqAXBNVTmbZuXkRxJ4sCPBmM88PxeSwiR9lK3AUMrwwJGHvBnbJq1b0xsUGUDmn3jEREAfoz8fLYxpWyZIyvWrZ0WSgGVzn8L0rIZYXXMdYkGeo3eyQn6AXJdvUBF63AcQt5ATeqO0LRnt4zJWcF9IOk4GJUvBrU4I6oYaqUKJQ6e0gvnfwuGwVzNLwq3ew1HYhwrdXZH7YNBW7Z6kP+8mi8i0mkzu/jXQaWw++cvjYxVaZfS4z09mr6cq7KH2QQJjpZzUtoT1jwqI1mu/hu0YQDKSbC2XFTtScKAqcjvf0O4cV1SHrLsbgTs9ZjqbIbE7Pm1/qrvzCNW0ysq5RG6iTT/3fJlRbwKIoF6jfH8Ad9vVkJzD1eVKgz97gJ8FbkIdP9PAIPZ44t0IjhE8uHQCyzMNvrJSuDwNVHq0C6Erdhmjr81OHG1yLWzPg/wr5/2pAD7iBFYm1CuU7Z1NqkfIzVh5pmDZTFsYMxiCq8Q5lOYIWx3T/9FTO8ZwqLsu+dXXfl+b7XwkbTnUKSMe0ZBfbJBJF/QE3X85seZ/WtpuLC6fg0iC4Z6OmqE9dKiRwSXFmFGc6FrMJaOvgL1rJf9jq0PfaQ/evQ+dQVoyLBkSiQahVD3IZYbRBuqrs7Jx1nypeAUbb2ukX5M5bn3SOKUfXlfKpVQWsbRKKlc8Gw/fBAlHF+6bLhW4tZ8mYvhjbjiiYKjS0Jrl0g4QgB9UKdDc5qQgIpGSXvNnOUBvG0f26stzYGrdy77aDcUESkI5ujRKAW04Vz3/n/ny2tZWU1ksOLIgZd8dM+Qc1MU9b88xSZ/hBHDHA5ghSpxTcjUv0G/ce+V1oQwwjnFJ9T0MIAxIG2D4nhPZKmsHJnbT1SRncPupEnQLMqX38KDmlpYuAi9kejJNaUZs2W
9hQYlr6yw2C9cAQWaXcuCdudkVFRTRv3hyAjz/+uM6v3759e/bv309ubi4An3/+eY3HpaenM336dHJzc8

AiApKcm2b/v27bz44oucPHmS4uLiKlM7apKdnU14eDjt2rUDYOTIkUyZMsWWJA8ZMgSAmJgY5s2bd
9H5paWlLFq0iHfffRdfX1+6d+9OZmYmiYmJLF++nLS0NACcnZ3x8/MjLS2N4cOHExRk+TsPCAi47Hv
WrVs3W4IMMHnyZDIyMgA4ePAge/bsoaCggH79+tmOu3DdMWPGkJyczIQJE5g5cyajR4++7OsJIYS4O
kE+7iR3aU5yl+a1HjO6dzhFZ8tYt+8Y+wrOkHvsDL+cPo+7ixMers409/es87gMkSSP7h3O6N7hlz9QiGt1i
RHf+pScnMzEiRPZsmULJSUlxMTEkJOTwzzvvvMOmTZvw9/dn1KhRnDt37qquuP2r2rUKObn4/JOZLjjz9
m5cqV1xSvu7vLmpnZ2fKyy/+WiwzM5OTJ08sDG91NuVfHUWqVB66INRTf+W5LDy4pqZR3gaWF5lxqnwIdaHyuC64o/ScRtP5J2sFJSyvnmTHT9F+DzGhM1ULRK4Vvxj6hpxiRKIEhkgm+ybHcuLV/qfTe4vvJW5Bqnd2rnb+uKFqtCQGQgv4mFb3uGnvpPAPzzatSsb+0TJT+NAdb/QbsCYIdeW3VLq+ohqLSwlmU2TpBWvOqsT/XfMDXYN0xGyrHlMb/WtJdRXKCOVxlXntlb30ekirLa0YWvyqUOHlF2aP+3HvgodfDKh5Jwl4kB1NeWIfL/Cm0llz4PuLsyNIqsXUVGCZNa+VdnRuzd6AVdc7E7qm6+T5PxdIwcFPm8eB3M57wzbgb46bAKV3OA9VN1kydqFqkNaJZVMpe0OTCPfVJaa1xA2BRPnd1vRsKxh6Mx5w5XxQmZx+IT/vSQD9yYPEgEi2p2/8nD3w/fjy0TWYYZhmjT1b1p1JTB5R0cDh5tEzsbCdZVFb6FwjuPaumoS8sZLYy1dazRtSZsuxWy0axmrbzbx2T/ZXRX/P/XMadR/zTsL7HBmNt61J7SKFKZvvIkX4bq9ftLVdkZu7VFB51aGZvj3X4Fb5WzItw+rn0Q42dfWcH+eu/n3WNgeEEAXqxaHcxUbr/lkRIR+EjKYMvGV6ujXrdCZhN/H4sMXr47MR6l+vf9ek0U9FK7f21jT2yfSlhH3OZvcufovCVvQjPmjNg3JH4sFwFwXI8k9NMAEQC19i9vUf+3Bt0IeaRemzP+vVh6RFF/afgPvq0hP/SHAXS/nQhXAQavlMzuAHbHtbwvWWrtC+T3AlvN/mltCnO/aoVBXhCfmbtEHf8YB2iGRQrRV16H0CtxwUoTVjTmCkXTVFhfv6Qr+bwVvAkWIP3zgf4UOkVgT1MvSj9P9P0xi04bWoz/ie/j4Jr6/tMtrKAG/wbdRWyyPm5Ebrw/lNp7ON03TvDUvXmxiE63O6t7Wxr+u8zPnGfz38bTd75FO8uSvU/Wz7vTePQuDzx4eBUpw6L3mxA/cWRxFOSZGfy06hFbwd1NeSa5fVy0ZLKSlJXLpvSvCYTEFCeoqxHVEFXrPy2RsjN7adXqIUz+rFsthUxLvxAULtMWdLdlE7vYTQu+/Up20yzdv7F8Wq/zzv5wEyq5EWQ7wgfBLTdjsrbvbnu/zC6V+qfKZNrPkb16dW9vfRk4v7QfLrHu3cGyHtZOt6yteRRwflvd1wWbJN8lR7GDt7Mp8ebfzG/s4fkxuHiYMkXl2F8Z8m+fJr97IxCPY/1V7r0YX3pXpbFnbJvo9cYrXKKLL5wGJ0I1PaHQzhlk78t8f90anX+ojiqc/h8B8oCrjGf3VZ+V+zz2xSzHqz8u7/OZ9Q4uvt+L/pwMCxRMSVi6sFEd/l/Xpxe+l9NfyJt+OuHqmuS0jJe6ZjzvEfO1lPBC/6Z+rvXV/0Jq0NbMAQKRF4TCI3FtkkFxHvmIyHFU9A83F3Ms3n8iaGncKCxOHb5sB2pbV6dMhxyE58cuhR1tvy+A/fwB/y0hcvDPj8ekqRszSmdbTJvXj8yvBFaWW4dSJhDwdlhYx6X8uUMi9i2Ntd99Vm8Pb8ypMlOrOf9DlOPDEPs0vYmkYf6Hr1+3gP8PntpLGDHm3tPprLPf1bz/7o7hmVbY9f/EdsN/7KKkJ2sHgfvuZkpkmItFOJC+PkFtjm+9QL5tfVBIPqAe/jPfpqqqTdRKSAGSS5QvI+bc9ytf/rTFQYcN/gH8pEQNiFd5bOqv9qvhWTz8dgAPSV4aV/BBIzDxMYRzFmJZoIt1dt21l6ELY1Dx5R8wbGAOVn5ZZnewEbaYLbs0lM+9+rPwtV7+WzPlHGrBVRqnNGpf+EaZr4fdPR7Ep5HH4a7DTfRj7a3/9bQ5h0EcUmk5JD6kFu2mHjxz/wgTbvZIrLZJh+/xszvbQ2kv9iZIP4JKAkYPHPPT+77hN1rT1zf1nfXQ5pWfLmK0LLhQH0+Kz+mtf2dN6GQrnO7u5lrMIZJhIzZWhn+Hj3OiJG4Zwtf9Se76/t8c5nmgOcMyFgAQVr7FzPxLGKSWrgvMxdb96Gk3IJwYJ3sZ88w0NsIU2L4FAc8mIJ3hLVZWqRdBZ23H/yyYhX4eOjkj6TwK/3dXyyoRhm9XaDYm7MBfqr0Y2wwS9T5ZOfPV/fUm1lkRRVi+vKfqg5DaaJsGCU+uZhs+q7VhPMjfdqKJTUG8dTMmI+B/Fwz6pl+HIIRpfSazgw+O1z7MgUyZgCUZ0zPsWh5yO5L12rsCJDk/P32bs1d7e18O0kv4PaFv0lmJ07+M3xuA/bgVb1m0bd7Pq8DZ5hU1wNqrCv7kz/OSxFNoLcx91nNTOY4yt86n4ZgV4kNWP3UA+fXHrr7YHoOuUGyYlyp8B+uYmmGj2h+7Ojrx+VWbA8w1Lx86aWtxAb2YeEMLqfs3aT3ZI+vu53lE4fZm5v6aw4vuL6Z3sB4SoHzTaeZYN4eLhgLSsXOtI6/7f2w5w4+G2l21eN+XcgmBXsmmRcOD6d29J7nDGeD5yr6zc3IQnHbl4X2+nL+t1F8d3UYYNaQXO3Cbwu7zkoYqfAY5UsptTZwRkL3tn/qNvYTKzYMDaxW7hlZRHHQkdR3zPmRbJZrM3iLOTybYXaFYTc+B5/Cq9XtJn/4wwTqdtKD5njaE6z2OiysAi6XqJTHGv3vQV6p2Hy0FTAY8yzNb8iAFe2w+jCHcRrdm8PbZ/9P7LTrHHNLzTr6IoI6F79SrdmKzbRC9qdgQeKuVq8GBWUGS8qzN3v39C0aj7YiFEUw5kXcOM2TuTOLDj+b9xEOn8dRCUeLajCXuCWjv0kb9/hg1ddTPm/P6GT7kLGFRhr6XSMd52uN0TqQLqvUtvaaV4TVAiZ1mdV+2jW2mMvwKpU3wdJSR3bsa4zJDYQ+4wGbgjP/UwqZIOLGlDVq2tTnK0CHYOTbhn8GSUVAL8mQJm+sLbP2j2nc+dHlv3ojoXq5HWzRWxT+ezHlJMKJEbRKKRdkpdXzNpI8CLHrQjZX+s9fStjn2Q4qsOIoHgK5RGuEuvWStJ71ftbyRZkdFzfQnGpwdOSLXKyz7QJb/KiMjMi6rSCL/+XlnD2Dp3rK6bzjSEtK5V3dm3HrtGNDgr9QMz2lbk8+L8MCJn1j56FdaJmbCMVw4IN6tN6uXIGQ6L7S/+e90Y2F/qMBGmONCbN3fEs8mQwCn8P7n2rPbn6IRtHA1IJzaaN9SSY1P2fUttw8XHLoFdQkMd2ydmiGn3/xdoo6N4Fs4/33Xa+nNcjVsR8I/+ODwXaBuZEfWXmTdj+hbu5Dg80uH0W8iC7QeYR0u6hxpX2YNMGTEUf29YLlO1nQlVi++ffAwW6Uo6Y1nSwA3jVPGm7CMrpjSsLqZa2a6v7+hdKT6UvaN3nU8g5f3rlpN7NHnyuIqnkqB2VIvnf9RHSB0G8aFJJHK+eNLg8NtFHtgjexPIpvFMk2MjodEXKqYuvltHKXyHyaqMnK3/O3L64XJiMTvdJ7zvO0a9Q1tXtdVsLdhVyoBvg6QwtbbyZyJaNRBpV+UgryLPwN2BU0WxAYi8aNPD0lkudhoTtJaFm71BXhFD3pfiCBYAUP/hNDr5Ve3TdzvH+Fa9fujT8xRfJO7bmZRPK3Q8w+ab1A6GSL7cm8BNjdPvMbULMk1tqz2z5ZpVjdDUcPMSuj1NlRbC7amNmZHLOjK6T0WFLeknxgjbSGP7uQDGYN+Tr5WfBYPeX0ocoqrswP9y4VvzyW1+AYZr8nMM4SWwWqbSwVqL3Q0/d3hAadKEYyGv+KHbc0PtnChe2NXGQBm0jT7FFM9jZJ/Mjj0ZHlc+aF5ylNU8QL7UrqXBS1lRJSqNoXFn6XZbEKqEkdq6HCOsgNf3HHdqn1+jIn5zZkjOLvGRxnZ6/5TbE1Qmj7j7C95NpR0e98rKgyQNq9/qHFBSAm2+eXyeE/TIuZr3ZzyjnNN7OCzdYvwAHbP+lStrrVqRAcj9Y1yFjiR8LTQUyuYeGAUXRN9vFqm2PqoHs5kTxPjZAIt/5Tg07fOjF1nMz1ejjU1I+8yOP8nBfjn5NKWMdDMMIUmTDEYEDUmxZvJRdmiMcbXLZR7bC+zuaNXGYK0Jwve1BX1jOh31+HMt5V6/94xTpY2TYOd8tv5VPQnmWpmnR0N9+Ezmm+HYu4+cW9wpdEvXXsOKxUuhxhXkx6f0JNSJ0qaD8oJcQAdpIp/TAIkOYr3bqj3A3u2Kk4rkEmxcXwtl+n/cNXAe8i4zNvUJ8zNQNO7cwdhy57ckqSgEVeNKqVMTHMGsWqGLN7yoNQq/0BlFPOMkMH4Jq7E4tdV5zGhSWNi5l+jB7fl1TlvWcBbdDHb+vAOdPbjSyxdlZdeZ4Oq6fZLRrmBZwpa7AeT1KyJJuD8FQLSUzDEldlrW7/jBl2ZCCqmeuuQSUfVS2lcMaFmvLC6sthEoSVc3V4eAUKCw/+2c9oGSMRyI8SRWYuHb79PUYJLoGfdmAUFJQYk3aIB6z8WwrQeZwhz04rydKdL8zsgXCBYcLxFQMmJ9FSR6YWvTgvmZ+zUm8R1CL4QqyM+b/bczb5Vo2jc9WfWfWfWzLcDW4yZhrsvXUDR5Xo8Ln4i0uRnoJrLfrC0oLXrCqIxLjc1XHmj3xNM0q84zEOm5hIggW0FyFIoUJYGNWXt1EXpqQ/q0V9o/UdbRyxZyWQg4d6yaIoPAjHx0/ZSS+uVGGQAA7//+VFUMdCx2VLzxRrJPErSNfs96yV2e7CMk0m2YYjYsufNsY6uakNYxfCA2mYt4YHt8mMnk8zUU5VSA5SnX3zytOyDDKFWbG5klUJzfxqNZ/tqVgU5/eKuMMPUH7qsCtg6yFE4yXT6nL3UA9nOQHMWp5S7t4rgBSaJZE2aLqWWyuWMAyFVZ4KbhTyvYbYdYOCD4IlZN/ys9l+EiRb2kZ6Y0IqsvcOL0+FJvRe7YgeEWiFqSLS5RMyeHsOPxoW3rxhKG+TPr/pYzUD5mZqQpJAL3tXMLaRfyxHlMBJKk94i8n5/XW6UrBSvwT0GGllSQf4g5awY/ruVbU7nqXK6pn+Kn8M7+fvvDcSRAO2+lCdhLDjXGnXgLmXVVc0gy6hqB+X+MpXqXh/zGVI7aK8e7DkI0oP/3fjR7uGlYbE/6Db8Nf4r0VLFiR4hHw1xhB3tN/MqqIK2e0ktHGoSYmSiT5lMI0I7Wv0fOsNdZNyMn8n9Nrs/7P1hu3/9L5ltTCN4mgY7zUp+cc44Z9w39fRtAF0Cag/MHBZVc6KrQ2P/7+nIn+fpfeYSpzj9L/6bEqvLOpjZ+ZWnfDPKyNe0fVwt3ziw8U9TFxnT+ApyvwcTadUlFmgmDTvoXL7mOeeC4hPKQlBJtQVjhD8/LPmUEqoVT5Qk03PqwNv5y6VUSW2yKnpEF7sVFYsgFYmFT8P3vOdaLKMnYcCs0FMV2dhWAZGYzpFJKyEQWnaQOiKDx3jtYUHEY+fVaG1YDo8gQ9bxWy9W+ydGRB7uI8U8gqY9Fjv4PPcr0uMEqIa7zcUJAQ8lyhXq80L6jMPuK31Cc1a0v6CIH9JKyHPgjldDqBgM8jtd7JfwZyj9CAKSMbp7eS3dL7ZgpHD8MZqu6z/AB0UHf8bpoD7B7XoY7JrJ8BX/8jp/bnsFEIBF5Zjjkr1fGd6HLUlIeEkdQf0ycc7/Tk7uMN5e6oJnNlhIKNB5ICM1CMWyy7xO4nHJIijqi/7dmsm+bY5ZDqFXNH0yS0Q8ryLk4Fw9vF2umrpQB/vnNmq6Fsnu+pWOzLNWWoZ0eZ3Fo2iBY0lSkAu+EKXnXXwXtV5X/jYp9pLjxTqTI4QYzBWC1Tdz7WVRwK2uP4jYzb3U3UQfAvKRA1VbJEqNG1SmEzUeNzxlS/zW30m2nKMRNH9V/bj9kNQZvrb7WdbaXn7HG4Zu5krCBc+QlPw2wm5TR6PfLNkn6vb+4eplZhg0fRBTEQyyEe3RWz3t16HVOlBTZf0h9gGSVnUHH5pIhP9qDynJ1XZKFyQt1SofDYvTW7IDbxT6/zBrNQlXSBkpwHT7OZcgWgeOjXGsO5gu6E5rLCOfgoEExtYN/bCcqiiM+rW+S/n2UW+Osi+9e83a1WbyQb9nbZMl+z7rG0m5PNtpCK6LwJCS7mTwbxbsvvx9PeRcpW2XUzQ4E3Ka0ktLkHRMmtCE/B7CaHZ1/w9I5Tc58xwI/Q7YT4WhXzQ85y4GT+CW7vaF4RN9Vqpl0MBAkYOG2fA+BpVBeQvl3jZXGRXCq+IqdRG0HKXzIl8FW3p04v3+KZYmdWf3KmcTpWCt1Uvcxsyeb/HNPgJgRj4VzmIQ1Ue4jMQQyiDZDmbpOaM8Q9ZeFDwXnUgfZ/z/6ZAAUJlzIxr2f8bSW/X96Uv1Kga2qeQyUh+YOW88A9/VLwvaUvqPeDsSnSa9jm81M1p5FVcdhRfXGpnHk6ERqmS3z3Jvx3tfLQ/1gEvsrCd5jx5kxVIlzqGx/r3L/uLL3CFnEoyg3MaLd7bfpjz+RL7SVNGlTmfYJYQOH6pqfzZNuv0PbJ2wnXyJX87q5MtlhxJ7xXyZ/GmI5hJTO3RmGJ6wMSHUGH+pgs0B7vaUXWxKB46YcE+7D1bi/eMW51n8vpgY/k7rv9UWQ9UxmLsRlKxRPaWK0SyQSj9hnBVZR0AWkPXpJtVn1bl9Q5e8tzeDoPmKp4Hn8W0IC6y8wqfOz1Vb+kx5XrHPMvgG0sEGg1HfkXUgDGN0ZEOw8bX7d4Od2lkn8PWc6Uv1E6xZfLx5j4Gz6+HnrxLt1vGYMSWhtVKVYfV9fVqgCKrKDwC0rLSL4fkTSAyx+I6XiJ4zH0y49GuNyJ2+LVwTrfNf0q0KJrKrQBXMDN1QGuKSR0zHBqt3PEUb2kDwrHT5fOu+WY+X7ZKPfWeiVJI1b0F9BGSr1GT61ufJ/M5RGqbMkyZZk+vsmZ1MBYC6nmsUjjGMYlsYKQ8WWgHv3XXNf6xeNh+nQ4OtZB0l+D/WPWJIThUT11PtBGbCRoW6PJHwBVVN1AsHO7ULh1ZyxCKMxdiU/fMhRP2CzqN5gBuvvlhsdrTrBRmIHkIMdMpsL3ZUxT2G0uqxsEZHOCqthUj9uWfmDXH1y4ynY9cp9B01sNuW51bMeEzPSrlU/S1lVcfqbrMHGS4i4kR7uovXCnZgSV4e0BBXD1e1HoVHbT5hpAz1J3V9w3iDQl2m5t/m/yLuXQF7bYJUBp8BJLTqMYaHIZ93Xzaj2eyiRkzMMFYpNhk9b2CQR3P2kfyxMsP0/fEVLhTSVs/FVPHHdJ9H3yufLWhgbaoWUqNz0JXXL8j1ZzLyNKZbkEp9eDXCaBBTxQlwdvxz9i7j8nfP/I2Sxr1D5wbOlCrbLm7/mk1wu0Zq09RJa89s0vZBZx2ZYlV4Kr/PlLjytMvULR0vXw5z+tAqR3JtnXUrLLs3x0KjZwbH9bXlpyaR26amZUx0bu7Hc94DRe/bwPDwOkdM3zYrB5UDGHnH1fdQzAnCH+0QNd+m8t2hQ9Bc7jP4mJxVhOcfUSqZbxthlqzDhl8sMtRWqvXlS0D1Y+kPqBMzW9bzVPBVMNUcUnfSpG/XVVgUdz4VmVWPS1FzkUhYNNUtrsiYbAl2ChvJJsTczoQYYsfGjmHfDlWqF6HdkZSpC9sl3dlmPJ8CDFy1dtTqkE2fOBvXGDMylxfjLVdvJBaP7hdNnSmndERzG2jHuefHOUU4qhyS1f4VeUzCWGw87AoLr6Ff0nRtRjjU/OM/s6H6ZhbhoRyXt6MTXUyG9hT8K8iQUGVkOJRdhc55d/xjT2K9RL2BhsRb3OPBA6UqKXexW5RxXWCyOlDBzbF3wgN3uvPdDGy4xNT79cMF9Xt3aq2U3ESo0Qw7ygVvFnUdb+WobxGK2AP4d3a7HNuPSmLX0EqFsLmyrPf+Pe/IMRJEWAfUxMH2uXt9Xl1HDu2DqgWfyB3uflNP49SU1kS1/HvZxjQvwk9Pil5kg5vRHDF5Ta8ikSszoSNMSTD9bQB4O+wHExkTpZtuOx4JxYLYfcAnnOpZ8TXAXSfOGq40kz4G/xTQEqlM4dv8Zwhn8V3iGJ92cfvZzFVvJhZCBhj+yuHhKYc37dlzsYxkN1iTv7K/yW0ZDR3LsWf4ZcTjrnXE0RlGV68Z1Ay4+8qQxtE1EvCnl4Zfeg4SJV0TOAc96lB0UFMdq+l9bbM6N3WJJw4EAmkjOlCv3Rnu7Ou3H4bWZKobuXaPnQ/bkTP+mVkF6Qk1f4l9O5a2Bx6ZXdeZ/FZ3PNX/eJfxxXCoypRIGm3uZNJp4Kq8bbDx8Sw7Cxyzg8ZVUeN7HlmZ9VJxpmEUPpNcvGTlWQ9kPJo+ubvd5iMycv4V2tZ3kLF2AXLrmAnkAXzWmhxcEcxbqQ8vSvSmhcpiiyy+RUZ6V1zd5YfdEDSLDUTwnNYqlvG21pPMj3K32c2oNZfQ2e4a2Ca5ZjYDwmo58oAzYlZX1oKo9GQ1M0FQAd4Kc8KsA7ImPVJhUvVlPspDzkfJDPwnDiyvv2Q4gSKe6LdWEW3ZyNdf2H4X9qiGDbZtxB4ZsBN3Ac/X8XjqNrsn9ZTzwsvYVYahWJXXhKRs3EDnd6R8kwN4Va2tdUwLuRg/pVkDKvg9EZ3gsA8CmE8ZA0Z9iGFx9i1Sdw5xPRQgH0Kn8O8B+xD7S6vnYL8FtcLRTOS6ZLm4pVWwhUW5g2mL9JSgBbBJbLsXbNzHxwwRl8kc4YSNXcLW8PQsf4EbbcjN/RWu0nKjeVV4cpqWNoopbPaQknStFnhmTrvJNvHYgXjHaP9Lk7qnRJfJfUYu1l1AHlO2XoWsgnIOUiSBzU/Al9oDSKmFVqVlDxYmBGtGmdk1VBb9nfnz0Tu6RNDN/y7m2PfzXg4a6DqG0cl57pvBh3XMnN9c8Lr+FkpHUB3sdLwZDyVAJ5Ymcr3bH5dItvdNKK2C9MhVsGEDNy0+wgSrWZ5fnHJxb1bmQEdn8r7p5gcgJ16TG0TlU6AHeSeGsHT2X5KS8Lw6W5v9mJTN1r8Fq7r2y0ST0PmlfZa2YIUbkYjYTnsb9Qy7sHH3NvL2YL02rZsDDUKv1Km0H7YGLUi3ZdyJm30Lt3cRXT7K7HGX48lAWWJXxP8G0cN6zpGBHEwpCJAWGk1oE4amAFVDBqGF2Ptno4HGA43FsNffvT0gfUY6+fqzLpuvnY4IDqzfIJqNfLQIzs76lhp1pqgCDJ5Gq0hxNLQ0A5G83xTWrpnyX6rppdcHhN9DS4h1VTKfJebkXW9jTfM+XIDUUnN2lIyKxhIIXP0qT9WD7T5GOZUOWr5ad9bVoP1kWcJfR5IMI5vm2Ve4NB+UJaRYUPPXvtbFfLGrTsfMKYC2w1Y0iWWsHwkQd07BESXr9oUF9uMAtXl8X8Bx3u0SFvfSj6sYnCvO3l5j7wvqpGi4A1lnmQmmP4J2cH7ua1HKXgc56QKh9Xz5jyybq3rHJBQX/ZXyDHbNU9YvsoX4ty5UmNoY/NdP2UmQKDpKWbnN5+l+pCbiy0vwAVmvn0xdvSpRw0ePb7m8WWS5Zj3gSd6B/7S6MBNfJ1glNRvohX1RbMmdIF8X1BFDRN0DpXKJdrNIHa5ONpZf18YqUGZdT2H0u7vtySqUGZT3TWNCjbG2Gyb0LRL7OxdJwrWpT2FM1tJRmdEHbvbltGsOWoP0EUFLMUb0bHXWLIvFbrWGsQ/YDNBwxTfFM/k3HY37oFY5bSg4OFahCNmkqO8o+Qr9EQL9hvBfqUyAtRfgNy8mllGbeprvHbU2mH55Jve4nkTbP7PdcT2LSdoyH3g8D4n25rmZUeeJzRcmC3dwzHjBQEJjTP7kLDXQs1G4b0I0V3xBz5Zwzq85m5zvtORFoV8lnw4f+S3DndJV2m2y0g5kgavV9BGJmz03RUOI/j0ryo9lF0R9nDRx8wSnFHeRBY6hl/ffFTZr8UXSDB4jNPGl8xlYZ1cMNhk8lmnJP9VUvb6R9ygZ1SESz8bMYNeBStqkVUNSjg55Aoh6edjAXA/dXH1sRM1FuDgxQVhRiN3bBFQEYNmnbaEtmhJdOP6gSLpR+ULu38ctDMWRtQt+pwwWZUwHNvm6LKsEbpzTMe+kdJfJfvN7mJPJSUCQ9t5OPGaM7jtaCWVq8aXLnSt4K1xI8rbWWvNXTuGHW2gfzh2/+TPSR+WIjKcT2Aa4uKBUKJKMYEi8Hs6kJDWN+qXDYkDMNU+6IRbV3iUmQdLdPKxPEI4jsIDJRTa3ZFyPp1HoPeWr4+I3kCVXuQXKvA2WQLS7Lqc2KlRIHRDRMsq4NMDGrhvYT8HdKhnZKyHWcXkE2O2tGQD+wRYzEE6LwtbRCtvdGuK/4rk0G3fB/nfXD6GBrEZHdbb1D1Y8Kxsz4gFyxb5NHgDbH4Ll8rnlgm9xX2ZXtZqXDlyeT3Ij7lbfT+EL0YWFiEVXNv6h5U8Mi4JgrKCIVUCJu0KK12BP04LPiUhaIbWfhaH6hCvlVGMRdlZ8ddLWETJxYkvR1d7xrvZyPjl34KJqJUb9k0qcZUVxBdC0SlHqPRx5ajw4DLxlczlAXGRD7zjx0dOnh4i6i4CPm6HPW+4tmDbLkHVh26bCWJhr+2ZqG0i0MlsKOHcGJzUiqcXwwRhT4dsQY0G+JtzpzH03hdeuo4FnvMdu5kfgKkEbcSSnjbDjyBf8CbLgv9q9ACkdRiwTfOoTMq+yb58Bdk9lVkWxRhlPpjX+9RcILeJfbdlsyw9aaYyfFRzJmeFlZBHrijK7+Jq87HCZpwvaGk3dHXVRLO5sN5HWefUu3qYf+tWZHdPUYbP9JMbu6o9WVqR7RH8oLZfX6WF5Lj6hkjefdrG9YXVtlFvE9zfiGRfeAB9YG71U7ZnpVQHv6TmUBZDPO/bVbOlfVGKFlhDWOP3dHiMdtffW8q6/9S9Tl8UHFv9Rlc1Hf6O0xm5Cq2xzDyNUVO/0o0sGC9utTYkXCrKDFEYmMz4uyuX2Rj3t9wPmDfzpszp+wc5m9CyYdg5lhXIzhzv3zbaasz8q0kmdFmyzG/b7e1bPaMV9PXSibPRzTZ9fBFwKAUHlXchLy7mmN3qFMCudi14yFZyUqeqNGT5G4s4dVWPx2aWv9xWuJqdyGZc6lLW2Uo0uvZDDDsHd6X8znSONNHUU5zMBytQ6uPlnCTMPR4ooXo+GW1Ckmm0jLXQs6jvUs38sy5+cm3jvHzfMFJxZMTq7U6DcvXnBUQp4ntXnEnPRP9n1YXOOZhJhhRy+LUzEOXOVQSrBmm8Dn5Xnd7L8oHN3FqxZ8RMVh1HRx+ZKm9LvwzfXnmYRqwGBCp8C5NMtu4JzcE8DpMLvpyYG5Sa3HvZnT4lfDkyaf8aVUbmkmKC98/h+bJtkS3z6fkTK1xXcsvZEdJVuKyH/HdIKqSUQLbfwkHt5lt2ekQ9Zy0k8cSMj8W0vKrVnywYvu8ZuCZpqx0fL32iQDhfylz+8MprVgHkJrKmH8iByVYfpJBkn79i8gyNbz0WNhOmr7y9xiyjrpQ+gVF9q1o3vHqKyWbXFI5fwS+HDVIgKf6oX1J1zx2CZBJ6GOpiFZSHvAyMfpW8mIuAiGBdPDhu6gTwq4Vd0QpAgcD/0BdIVaPuiqv21wOu6qH6ZWnTuwQZpLumvEzJMkpAoeCBHZI7J1B5VuF8SmRzgYjm8W32YpA0bMgo8ezp4fdFwVIuAuLNp4SP6hpgo34ETfAxgmiRPUGKnqcWSdyZDNzoYHVGLbVyRZGzExwnkjAO2nN+9KrzqKB3zgrs/n8rlXbyYsXBo6urM2CrSzTkWZ3WvMxHBIwLVYjrRlcs3IkSJXm5XIXgqq1xJG7x7O4d5DEPBm2wjZrvQYhPVkS0PSrTh9Xs3o5Bv7dQcWnDgPofzvSpHRqsMpUGu3hnm2Ycn7Tmp84LZF6qe0xCG8DNRTeE1ULHyAZ+xMbn0h1pVAAjzttRNdsRBgqmHSlgvMkRkI2NI01d2UpCVMPbMBbZULgQ0cfOmZDbLHl8UO1fIrIP5ni9V+gp/CYBoNiOs2aRL6oo86OkyrnLTiXCgfAqaXwlQiyuhFtEdgDI4VO4QW2N7cAL71NpVWG3xyp0pzqHq7NjJF+RgHy2DxcY3TMnfiPdy+t5T8kZkpWE1h9Fhof3whzJuw4DbW3v/4K/1b0Hp3M14VEqQWsQyHTa76ESXfAJxBa/oKxt15xbAh5yGm6n4h88zHR9cW4ya5rgKR/zNKa+G5EfExPYdDBF03cEIhN3RbJTyytg2c2ywaphW7NMDIxSg2OgIbSzHYnPeW1pwj/n8Ryx3TL6jmiSUvSDNH1l9Ol23qEBFVnLOMwC9mWL9dXFlFkFWzDdqnsmXWfmt8Y9AhL3tU8qz7dmCDwPV9MVYlJjUnHdV5x0F03EuQTpIJXC57M/7Vwgkr9IgTU26SeUE9ydmBkjCGDmWgLLa7/uPUcJOnMMKVR11AUCyGVcnp2qNs8xhrt2GeGd6jsMzu1BaWlRXCHw3IOMWx6Jrn5eljHnx8KpuGebc8htvlkGPwMzPkh0SJjjW0Dt4j2p1ivpjQFPgzOeU1cOGGq/DU7a0mvKGb6R8ItZ2qfrOyqlq2cVVhwwHGRMYzB3TstrjaktcqNkRlZnKHGDiKKbOKfuy9Q8vZnZY39UXOmjUl5VmMD0aJZSAhQgw14cvbJUA25FI7W3rzOV6dSYzQ8mqkmUpQfjCnzfrPSYXTFfqt+I0dyxrMMxXQW2lsrvwuTdfknUavu9OEt47xudVT4KSJF4V3w+AT1e2thR3wBcrl5dHxCUgDgsH/OBHAzs5Y8xbztrarYWuNY0BbAaHT3I/yvx9eAnWRJ8IgXkEzFTNRxMmyrDmjvuMl8EGkoZUbJTSMNsaLq6B3IOvUD8Nzp8XmP02Y/n8uWvmmYuNpN8otcXo4RaH1zhFdRrJi2uB3WGdT8BS0Xj9kM3W2s8hJ/XkIpTqT8M6Z5dhNS1WJgFtpTfAXJQ5+b0GWYgxmC2kVX+I4Z9gJd1ndlYm0aTrv1b6QH0LqX7IwIiaxDqK8QUc89jJBVIjPr1FHgrxHT1AVvr8dZq4ZNl/9ZQzP5Ai+qBbO7W4x0PQLU7IdBVmTlhPCRJ7X2bFU9hVPsSCTjQgOGHcflz+6Kv7GtnqMbwWcNUFHN5DOT4wwjRnuNdjRlJ1ATBqJjhdzRr26Pra6RlnO1UIJQ3SAXsm69fEg9oNuPo/BPCHH6mdDjf1PNVOuvPvJ3r3cU0qdRD8Z2qNMK3pbWJFgvPM5nrdxGOvvGQHMjl0y53OdYWu/I20cvmMrIDe4rsOPeiYi1W3Yz3L5SSdXtQmLIIDQl6fbRDJyI1IRGdC3lAFMHBv7MXKBQj4B8i2NVvFbCqrPF0otFwK2rtPGvYtVKYJZ7QFoCwnPG4v6zvWSH5Dr10W+9XZPvYKaTl3E9i3pPdWgxMcVTaBKhaLmqsbLaaP08JWEV8OuUk9aLDeBzUhywO3Z1FBo4+N9qJaOtc3OFzzc1ffT3pgDofVohLwfe/i0kuJf6xqs+v5oGdhtWCQ3M1O+aUJVOgBnJdMSE/n3EFhD1kVuvDLe/Q2T7rhcuUM2ND6a8ZrgyoQSEC8p4/P9jG1sCU7ps3ECGYtf+s6dPwtf/pAVNQlcf6r7Hh/sfRk3VydxLwRsNm/ujk8bgkU+a1Rb4T43k2XYoJI+J46jbN1gnMbkQLIR9LMRt1Y0CjvQ1UxhS7rv3dJc+OQ/V2h+JNAUJTxupJTObzi4j6VN4zt5Q/yezVoI3LcN/ArV/sd9p4Gb/jAgWkWdoN6cPixdSrx8HnePXMm6hT6k9pnRQmpP5h8vUTtdjn3m2LEeC3/AtzLWn2BTNhqYa2LkmMHqL3A3pFzHwV8EtB2jVfnqwFM0ljqpkn2Gf6MXBvp8ne7aD5a8rfKB35TpFDqI7X11HXtcg1zNHJ/X7fUyAtHzQ1qoWyHavGMTfZ1xVZrZhO0FNsMaW3bGs3Sh/TdhEvvzDrvDi7pSB/SH2LmGQVx6r/ZxsU2c+KvW8WmQ6fhAB6bH4TKGT8a5jLwo5pK4AKVAtn0lh5vmXm0V93znr6jGaJpNhnl6JRC5cWQobGqG2Mxom7GcoJ2CfJNZBVfRKk9kDf4vQwCvTaQM89jHwgrmjhmEjXDHnSJWsbugPBSafsENYF0IAD9F1cNUnvxGz+NvMJfRp4RNcI0KUvTOWNF3hcIhUcJVzCZ3HUsTuk9QPyAoKH7BCSY9GvWlMOtd4FQrKqflaf4jyekRpQpe3cNEqptHyMadyOpx4gf/Rw51yXMX0Hc0zW2TUaFB4zZa7tNMf3T9HfnNRxv6z6AoEadhS3b5YpG7xSfRUCxvY2J9iE4JCPeCuzXKiO/1MtOdIq/Stjk9/1d3CgD12sNr5B0zGO3RRyt1aRV54h1nF1pdSOnAJ0mPNKBJ33/AvqeWn4KGGu6trT1rcjHxlEPh70odc8i9tX8MtzLufjDaqwN4u2QR3KK+iq1OX2oJ+nzC8Jc+oGSn/eTHzwpzDIIuoB/OZbd3nSUJBzZ1uXglHy86dVTqgzlknMCP6UsCHR2Ht3WNLmsz/3ff8LbNq6Zpw5ze/tcWVWlnbPOdl5m0/zN04cZBsvVW2B33yT8MJSL4+1N4ZPDbRiXzXlAYStzjsIe4lZRk1oAWqWUBmkMzUKN3Jqa5AaGOa4OT5YrVw0Ml9IF4SsUQEWKzq9gmAgT8GSdIgntBkeE93bTJvNcVKn8MkUPwZEAnbAH9hYm1+EOCfFZ0dbf4UtEllrIcDMRL5AoAYuEvB5GFyRNhZqZcFyIn7BVvPgTrIh6iGmSLkvcQvl7jYCMHH+t7ju4fAxlwIyHUrvNNl9NcDEJdUo+Drf1+S2JrJ7VLHtf9J4gF6pfeLrIEdg5T7j5DgbBVSUHxNJqj2dbezHtI5Mgu6aSGSnDhTPTK7s1kvk6nnj6wJ3tqAOGDnHzf9QC/0fa3ABDZeD+qrcEb0PTuEcCOqmZz9WDAlDDXpkYK3QBW+zNbUcCGKs+0BDqhCgCRa2CRvi0EtEFM6tgEYQp+g1g3qQs0jvobGU3ZLPsfQ6ml5uH1OE6yaz6UXRvUbMyeMh/RbmeuYK5pGzdOsnKQs51DCdNHmV4MoUk1SnTu/0JbNo8xp5NflnwlKyaS8eg6r+MGxUvZK/PuohM4N8TqfvZrn6V1Yp2j4MXg5OmOkPTsj49bFNHjX6aCBK+yZiNU7wsu8TedIBDSLHr96j1dtM3IpeOTZXUbmdBMjWe3wNB4dkTmK6FTfx8cXV5/J5z7+gk0VutdVT5A1NnSF5VoMOFdVRRtE2c0Lhs+7Y5ZHYw7P7TmPLFXKAdyl0vOx5Hhct9WMGMtqlvaVZMK2IL5X+SuM3m+WCJTbWj8AJvG6bSpObQtJJZ0H4lLbpH+H2/XxmK5kR8VWhm4f0H0KsSp8d2jqsVt4Nr7WbHDXkOT7aQb1wGwv9DtmE1UnMN6ZPttnY/3mF7n8HGH7z9a8NZ5H0Vy3rH93FcxNjBQsc/oyP3rBKNwjV9BgQ3VWuaznxcHt5BYjdp5x8U8VuUlX7a6L1ftfvDJ9Ag7gu6LFyUOB6iHrG9RRbx5zhwY3a6m8ky8XEapM+Q7srC0nnZYxEfnJWmnUGu0rvpuT2NsKxCP3M8pMF4p/LFQuWgzQYsvLrNOeEU+aabOjdSsbStb8zMhTq1U9N9jwj5iMKMlzANO6hdzqSdzmrRcg1gy/8ZPY2VSNxLbYa6VgpFbFBkNr7fOl/p4rgMJEMUMF12Yx0CoRKb6b/4MaYPKBEHL8tYP9lM/Eb7KxnMp9nS5rnQ/q89j/zK8GJOFj+dOTtMchCDDwWVphTVRJbsfm/nvyl7sN6aTgpPBeRcX3aRwTJJRSIqc5QLncYh8eOMEcNLXQJ7N2NehAdUdFOxRM5H5n/fU+njkN9OwqgqZ2xH+i0H3oRFEV6yCfPVMBpsHS04X0DBSXf/S1lEPqj6Xv1NI9aUlVYOQOkoq1JtxWiUVXUqkF9PeMj9/iXLtR8Z1cX/Rls5uEqHHi0l7L+e6C+1dLL6iqGHYZgTi2nHCkvrJ7Z1JG7VBOuZFHIvK0cHdHnEczZv9xf8LYYEJHHXf1LLUeStwk/T/Nui92vdb/1b2AEIGBBw+6V0WPvY5x+WeJeErsbUJHILVAkEQZrldjcsK6kNuZ8gsvBvvmZxZsjL1+DK3IWA5nAD9bY0GRsDmUT6Pb5DiKQr57glmhf8+5D2MN1fTL6Gl4kwL7cYW+TY4qDKWCXSB5V8d/ZcvFF/AgbqhzoNS5ZpU81+BMqg7DP9+6nLdNyJj9wnfZuO1CxZ9/pFwwa+3pD9NVFJ1ru8FctCs0rjy8QhqVwQxMI5UP/vkBzVZ/1POeM/sYrXiyKpgXDoxXm0qGY10PUOG37Mm/wYPk9RbXSqBUFT8zNRvZPIfp4PzLmWBmAuMt6RLgE8cbmmIfYSAEGwSs+aqfvYOJP1tpEwSrFhrXxHrWl5DWjMpoNA2Z7t16oomzULzj9JwsaW1TUbYEJS8tW8j96vIvBTqhGIg8wmuXjZm5ZwlPcTRELrIcYJOlPsNIeUn5SqThwoTgbXhAKeWvLbOAjczdrn6QxldLhTTjLQuDuqeOJhAHcB3jHL3UIGUR4iXLo9qw1wgLPLK/mdMk2iVRQrKtJZ4rQAkrx1dkyQdd3cp4WWp71M7mO97WO7pHf5U3yEWRO0IS9WoOuDZU3rNmTqaJqtD3cEs2dzMOsLSKbGTNe+9pMkbT4wC+o/9T4g0MI4ugAI10kT5nR8Pz58VKTQ/GWHcGRMrpJdRmRd5fVoBh1L9gvywZvrWNsbjfeQC9jKMS0dGNgDwJQ/Rp4dLP/VAwMbtEgVtoynq1w9zdE0ls91lTfhWKgvIjEcyxPBNv3F44+0lxGwz6UrrXhQt3NUFRZWqjqrm0nMu8rt2L9bi6KhXIsfm87YtI+sUttHbglrewmcGFdR80eTsUcHwJ+JkZ6Vv4ABRfhiQ0L2jJOBh8fsdkp18LhSwvBK4LHpqqXJKpgkxJnIOtt+Njo/NG69b2LwJPoyG4sZsA5k3P4SfWI3HcfXmAzLPpXfOFa5Cs9PoM2LpnHS3UdHlHsbvZnDZUjbFkLEyDoUnZ9JZMR3HvsM4PeMfRM4b0dOumO0mmmQr8utSKgDyDYjDFhPcXffM6NkUKCW+mx7Wmd7+9w9dj5XahQ6JA93awnHxZpsjt8GncJIn5B5efwTGhDIs3OIyp6Pcz3cOJPfMh6I7HHgQEkPlH6Kr6P5NCYNYXrnu46zaYWmZX7dk5VO8Tq+8GY1Up7RLOfYbwz9UkRfE/c/xt1TlQ8XbO8eaF11Z5v1OLyYq9XmXgQR0OJ42mnItJB4tEzJ15A44/G8eSYwOXdoJ9lOdQBB2FJCyLSjwdSqSYN5vPaA5DyU0VkC9lkzrMaAMbw+N7T7WBXDZbh6KeHdeQMOvs/bqRzxBbaNcFdwE1U77MwqW4gcbR/qzq+E5c/c/F4aE2PftG7B2VqYZhQoR2YAjqjjvsp9HZYTWb2f0n5t+ZIWEb7GWjb0xd6Sf2Tm6ruRTwoqO12k5yYkiFs8XrvF9o4iMlV01TWaXzrVWJw9ATiyk5+tmFE4ckjVrR8/qf3XwfcbTXh9tRxVJ0L5FsaTTJOYtC7wCgrdg1fH6uM3J8PIwF19VufEP5IBvR5wpJFdk1OYa7lbQaSGZCNnYtvTWb72Hz2DvxDw1VGs5lEQ0P5LGqwwXLSTPmmcmcjJWUdaq7WX9XM5dT1l4l/rU1Nz4sLKeXF6DPlWkbXJDbDUN0MCgVD6TfEZTV8bRe4SSIN4GYFvfgoAPIxHtwLyS8ZtMaeRxPY6cxGJ1bxN6rl3lgeEuoSZZ67g9pT6Uk6nCpcBUfG1Ymq9wLP0C9C4Gn3MSlUUpLZt95P6UL8JSAkbTJehakkm5FC9mCedJ9tahZ49aPLn0PbWnq2SPjSq8m8s1qspXz2BPgJ3ayTsbYXJG2JbfzN3ZNIT+bOZF0BNAI6dBSBNY+8RxezTjqsapMyyhhYBXdOP6VNp5TSl1gnKhR4cbC/sZvgXoZ8ww/GhjtFPpz6AWeKkvOlKvd0mPc2PTGBsJK+iztHG1WeMI1hHOcDT4Dm1JaqPOyj/jhB4DBSqa9KaH3j7L9dfBYw4Fb6qcIfZI1dv2Z7p+A+dK+7IIGObqZNWT7Cdk/+E2U95rnMsBvlhrMaGThsdylZQuvBgmVqq+/GFgCE6Gvew3j6/6IlqFtlWZWjT9Ei99+Nuvug8cXF6m7i1Y7JTZrOV9YZ0WdEt1HgO+2+PXHpXQNfdFfpbsnYlDw3SYnrcX2v3b7ZetUcdZ1ROqpA2Sqb3Oq1+eFUYf9TylxDZfaI9idGntsQbFYPwA1GUXqRuU/zjMhpzgU4D/PiXwgH5I0DfvJFAvbE4f6qtH8lP1fNTS8wgZw9QqsVkRFyTmdPMaGlrbxeEhXcrVh5YsjqNr0UO+s4w+SvqlSxYg9rhMvj/uNnW9MY5M/pRJC9cVH7LQsWfJTetzgDfi4lJRj8ix3mZnflxwlkU3L96U+YSGx7LyPzwtWb2gLnYiN/Q6AzQWgPAxavjoDZhvumL4T2w7IR2U8k6+qoJpOftvOx0p9uGCMsTxAKwZvuqt2XWEOfBx6MQ/VrvbdT62tRjeYT3LojVZDUOVuXFrSiOZuK5dgvfhZ/b2xFZhoHRR/aafRnflUAQ6jcxq9bhozWkkTrhcCkZyiFiDkV5PovtaAXyEtQ1LCuBwmNPXH2gK8idNSwIrwSkl+bTSGMT7dpafkmv7NnLKGI6ld76LLW6IcftqVhDMZv+UNsmtNQQvZOlKNgw2QP5dv6URvX2oNRv2y5DWf35SySMtQBzhXsKn1RSRkMlYx+XCaMkzdw5TyBzxsPRiQq6HvzI1IWn51QjzDu0vJoxnE5hJZw8LlW/R0y7hwKO8SlHbYtbd08sXKpCgvl2zsTXz+m7tcKEZvvd7ebTqmNqvX9mdYRHgKKz3UK0WtmRJNX7rPIsl2EVK/fE2qazImcq0TjSUHE/CM/RPQYmSQsw2wQWljQ5R5Y07pV04CvhKBJ3CB5j6tmm0qzrLnNHvoGQPsYl8MQVUNr7zS7SefNyyhTfBTn3cQg2qfYpAYBlyQLM2CtUHUa2M73jsqbFkQM6krJiNnbc1wIORzGrAwR6gbJlkcJTNe9GfSAgwDHQPMOrgx8nq/L5mGOG30j0wgxWi1Mh+RAn8LZV82JgMxy/87bIkd77SGfDbH9/w6SLhvp7O2QQ1NZW6XV3DKkMsqIuLQtsL4Ck7WkZFjTpm6QoO+LPQrIjCJLnS30RVCjNRvzupIRyvC2mEZDYO63R9OdgDbk+mBU8/QYdH06bxcI8l6y0bZUnbh4oyzB2b5U5Zqa/1DeSq8v9FbOsIDyOmhK31EyFMY51NHD7iCUZVC5DDSgN5L01OeOjBIyb8SBYkvcH3QPWa7gZaJkLWhPKxpRsaT9ZiyAH8dRe1A5ZeX2aGJBrb9/ddEILMnTE0RzKOgnR3cXVKp4RWMwsOn1xlUlAv3XV3hdNQDa7ypMb4qTkHT4uIvsR2VFb1S0o+qgMV9CxU0OY3wwQn8W2bz7G2g8nEhrFp/Jnh9YQzKhZWjpIEoyqJpTaNT4MvhGVBKhG8s9uYVQvrzgZDzMNTwR6TE8pIbb4qZZ05HmdTpjUvkTOX2TnrDxCa1vzFubK7lC20MLwlkRiVdNuTrahz4EjRF6eX5kkx6MeeBLSb1hb3TDtkZXXDBgRqfa6oUZvP8Kc4L0qDKWJJ0IPpNvm1tXIWXwP/fetZn6z2E4BDwJAPdOWSu8ZRWJy8V8DLljUW1EmEgnU9D0/eYcl8oXgSdMyfxWVm4T10IBhHdFzBtvwJq1t1NzZyUo6cO9P+oj6/2x0cEr6n8b+aIkzpXIYVHJAXupMqPkS2lbWtqPAA6WX6TeVUF+N+rUW8sQrdcl+C5uCMCwq2Byqq4ReYZd6zJIQAG8LUjIymgmOaWr54iS1Mm6LnWcZ6i/h3VdhzgkhGwYgV9hOy4n7m5c69lpuDpnf8a0ByiYNzD3ATPqRDIXXQ5GgZmoBV8G02oCzWkx2oYH8PPD6l0+7uH8qdgc3/ODfZ/Itq/TaDS6eu8Ut7Zy/v4Rh80KFOlJoAX4aLzffiSPrgBzbpkd0EtmA3HUjUeCxZVnjKWdbq5U20I9ehGqGdOQH1UaZT46LRsCeEiIGh21D8NaX9+p3rzd0TmxSlLZrKxNwvxHypUJyh3rRTHh0Au5D6EYHR4zzjB7jJO5u1lIS8tUmWdy8yqQr+1FZ+j2T0N79E0x7nktxDXs2lJOn/CN8whYnIEbk6i5m0OZ2o/CTpK0GMfUgNYo4Lgv7JyjiqMzZBxEI4zUUNmNdCs9Gb9KJsYd1nCjbVQFsYmOv0CIfFpM4sRp06Xhd0vApqczhovXqBHcqUyutMqc2nHMKWHHNXfy/hMz4Rr6vG3lg9iKlTyIa0Cy7kBL2zvgKbN12zn4MWvFc/dRCjCZDJa/AHR0wRnDX5PmvOgshJWPiAMb92K1GgtehJCoqDdIydZT3mQVbN5pPJkxRTnWR+xfhxIcXhhpRhxLf5VkOa5x2DWE2rqDD6kMGHWz+uXEVGRgaZ+6jO8Qk0ZjgUDOm6OH3I7DAIWp93iZnMQTsT7LkN4sWyCUfFUYJ4aMWJNDTWE0ZVq5M9BN60bcmWCBzXWZ3WHEqxAZvl9rXHqRmPMNOTJXzX5PEDHpxR6xZRZ00wr6g2PqXbXjz5BBdsl1E8aajOjeNWMtKW9JNHubRzLd2MrMGOLaDR7A4RLmG73qOoPnCoCzcDlHEDzxSfFmIObFXJzi+f8qyULzaKXoA1SmbYcNfONbe3FcP9wTDgQmEk3h1INffdPeL4NTMzDg5z0F8kOnaoIKZQ/xOi1GiCMO8oyPqkVeD/hN0TPvnNtdR7yH6ZrMcgaNEz7nSzMumO6d7hhd1FZdmnufKLvM/SAFxvR2TqsH1/DQ7Zs/OBKZG3UchSwHRLP6fzMUJ+6x4ET7A3E+/SbiyphcOhBI1yxgn5B5OOWrJcvXr1O2WQcQ9ExPdvkzmSk0Zzq9HKUwo2Y6xJt0s0PnPW07uN+uq8Y1p1B0L7hIHwOIm6ev0pzhqNlYrZ7uyoflcNP7vjYcyzoGhyNuNUuWpO9wbypwLC5ucymY5r2zpLOCQKBW9+1B1FMCLmpNXObXm5r6r6qAWlEovjvvx2t0GKH0dGnoD5cKHlNbtvAdc8ZjGaQ6nZ8f4mk0KbvTdjb2eHmcSXzCcJSgfafoN8jL9Xs25MqHmTDdZkQRGwdzwfOtqbbuKakYEn+dOgtm4P9SJUjuoZzCmI/CrkF/IJcCS5hODLEmM9axLMW5EFpjRrgT5ydvu/qWs8hEMn8IxNbYVnh14WM8u8DXbcWujPEg6JjE5vzTlbY7Gnr2Cd4hT6MrNC3tgXJ1ipIRk+EvPe6yCUHJRnr5zb3g2kKwmL1RW2OGo9xFj9Wd3+YdiI7e1mrVZ4jZiljEQZoU10Ec1gjfjqB+2ErHgVHdq8cTPEW9sbbwYtoGHvudwIsbp9f6ve58lrEzB3wmjd/LI8w8ZyWgcVzRRb8zKhWj6fUOhg6q58rpXJTnW3Mie9FRIGaGRp6qkUxKqi4S6CxhlEnFa3uOEvDKPeuBm5+ICBQf8Khp2fCZq2D4C7GV/xrV1SZRX+CmZk1fd6DqSxpL0PrajTDrdp5fbvxhHazk7mfpYB8oSK8mbrewFuuHcm8xr0Y8M4h9n4CsFr9nJBFo49yQDS3X6u4Bmba+cHjz17BK8W9W0oUGubX7NpBOr0Mx3hL12Xs8a1RKvOLU1KHHZyMFKMuKtDkZ/oS+nmxJY3cJrS6jJ0kEZZxV3PgFbnt0gdTaQ01xfTt5rr1dHMZdAwy77h5cMkn4kf4S6X7qXc7ZPP0G/5oQBmsNJK1wd7VZfcuDoxeH3cLoZe6SEWq/9lPnlI/8O+j5R9r2d5rs5klLUTRFbRiJGaxjZF9TFzN9UVaxUPCsNELGGRf3jbxjTpEcz3ZIZRB/WlN1Xtgeu7XSVl9KIOwH4S94Bm+LSRr2WxuqHMpT2QBb89k2xm4VTA4MHuDfamMmqBRSWMzkeEOngKxHM9QFUnyMMDtqC9KYGedvI03UCz/Vc9ARcgAv15FIP5oJEh7dRLQvUJfY3aO+AFUKUh6oTZgdUnLPjG8y2J9Jmu9kZSQ8o9ZYjKhAJ4hmkpWRnrG8Q4xgdBbtYgHxLDKZeutVAULmvFWO+tU5ErtptS3bmr27VgRjdd9Zp4QzQOqrOBuOWr9PkF2HfSkcVNvGF7wBtcOtHzNMdeJx4JyFXxnnwGjY7TIYYUh/6O2Qsi5DJBmdjLH9GaAtnQYMJ2ntTYYpwqHr5zWd1dA0M5eaA6NGtgRC1FoW3xrypD94fpnnOoa0HFpEqIomzXvpKgtvqWP4wzmFDQrR72ZGpGsQ9avh/2pD6oaDCvhcQhJ5kQRBYJ1WKbSLqmlwbqn24B2zy2KtNvJfJBAD1eNSv6FmaMV2ibvSdBmfuVt3B1yWmc8ZFubr9H2mdKCJsRBrQrn4nv/0gG/p5CrDbQzPIaoVe5IV1wUyw40WWEZ6N2wyqbJLzHVxf/rcLC9GPpuPAshkTWMxGuAIIRmIEEg4Ix9NXVxsV1GXMkJK53eRxKVQEmmOyxGzMR0kkwPjZ/RMz4ftE+ZBNQ+ruFVyOX7rlpVzfabDpbR9JMdTJ1pODCE8VUv5+aVcwyUf8vQMVbJPWc/3Jmh9zbLi00nvwGmvi0F1THeTH12KUzmGpzbqpQBF1IaT6nJ3fj46Xdvm75tmgrmvaB+fdc3BPQomchxChTUy+H4yFwe2IO6eP16ob4mqOQGGS8Xh1WCRcfbsfLjVGwdo2ADULPVtHzJfO+MZ3YemgEU2e6LQpbb4oC2Ccaq7CgONhKUZcFZR6H6a+kTRKyIExMRtoQgPw/jzxpHhmYMfP6uJTp1+v3xHZPF5gh5MhOqnrO2fGYJeTPsQjVjEcXzUtvNjcr8bgJXRCTEBDXJnIxWmAVIbeAWp/v4j8PoaBHbjH2cvhDlAI6RnBr1IXOWyE5QUT9PTsfwM18iUk08+a4fkYv66EZjVT9YbwDP0CYQIRERY/ki94kM4sN6WCdOtPKDo0wdCo5aLJaoYHRK5DvN1qVrT7ToR8cF6TsXBGLgQLIGQDItm3Jk3gJmUv7EQ0OiFNWBxXKYdJ6lG1KrMGx3KMPG2YG2MMpgVxpSZ4Yf1KClvwKRK0J4rVUFRKYZeVjuoO11IYsf9NFAXGEg4hS6dpo1EQzlnRkWnxfW8OMx9Kyd9zRTJCwO/zcpJAVQLDOY7LQGmm9KQjAhoZxJp9J5pZtwJLHw3dRXo7yAfCFTzNWXkAGdZ4ge2a6wgIgDe3sqfdT5iidDyFtQSnhaYc4pKBkEe1A5wRoZdpaxLsSOXqlXrKOnj4Ksg9v5sjqEZSJrIt0w7hJBPp8B8EqRPUsW2SMcy3PVe1aCpAgaFlFL6lBy9vkUNHK7CWNlAPF6sLUM9KDyUqtPtJAH0I/Z7EvOqRkHzEJvpz1jCyXNBTYD6WtO0rKJsUGMYhIL2ZOUw7DjUS1aeVwJHoFy9XNJJZs7i8FuYlRQSR1uMuGQzcJkrjRZaDaTZNmVZ6QmQoqYcwOhazyy8mbExe4LKWS+Awt91RLCR3AejIYXqwm1mXsLAEWEqfNP63FNvm93MkGHOgkM7yu6/Udf6DdNxfkUfpjwWR1g9kmFKmAQc78zkJ84ujKchTlJRNYaUafrawI5t7Mr9N16b9E0PwN+QDk4PFyvJUUtxYDA3e9kTjR/bxK0yFjJUv8fdhq8ouEC3tBQPcSmIkmwQhgsVWMhYx4M+36a2JgiVOvZoN6Xe52eK+FWmn/pd2zPz2kyQfmfnWKJL2VU3z8YGtqPfQWZRZxyNTLxAa/K6wTrpvL5v6x5Ym1ubyWwxZ7tZk96cP9Sqb6lp8o86EEyk0QxhFc0e/jwSfFnf9A7OO+VUwHSBijzJL/LG2vRRNqXazXdMepQL8CLRd/e3S3Mc+HvEgA/I7UzpE5I8xDRZKgmwUhFQmXoe5C9O66bA9APM4DbytVdOzLBDyRtwDUZUs09xbd6JmVdTaf+vFKzkSEIDLyFSj0MGqTk1JMhBdbvSJ0fRKYtIn8ml/zdfTRVLkuzKlF2zFOmZsfZZahQ2uOaX6YeFBMOl2MYbdl0pAR55S9TxxjyPVx5jJlY7klkFvnHKjpRaSDR5aq3X2yaXFjL6bDQEsPaCYd6EzMw7ykhxiwlKQaoSCA3gjXbEI4fMPlPPOHRKp4i6zfl9hLPVJHQCuwjnx4PoULZqWB7XfdL1LGSspOBUHGP4AgRLZnaK/ocWaPJbKgR6xvSCKEfXwZE2EHJS4oN4YUHzn5rxdN0q+cg+RHukKsH6ggvmSwyBtD/wLlXPI0AzgcnmeLJWiDEz3xn2WOezNw9rVW0d/IZM5YDvtAa/K5YCNJpPr0e3XXnsSG3FfdA8ts2QlkVoCwy4wPvFQnRX8yqfTC4NfONhoJYaW+3X63VpT1ubM1SmuvQhKW3wVn1IUW8cfbbbnAmJ8S9rI5BXgQlKEbsrF7sC/k9D2J0hrnrzbDgPj0Rw2eY15n3oJPmX/M3d6bn6Oi3uXfbpSbx/XqlL7zvQ2zrWm4XRbRdVwItDVkPl4K+3NWqo7hvPxyhHnbRwMuycyfvOnE3+4LMAPAnBFGoYKaVlqjQFpaczK/L8Tf5H/XPFCvKmJ7TmzOaNgxKpvxD3KwItzEbkOrcZsRJ8nCtyeVMKqdXNTbhnw3a18NgWCmnHlDAPPXaxeFf2c/VkjQOyhY2RqMH1FWMD8uCj9Bh2QGKLMzqFfjC+1eTwlTvMt5WptgQMKNW8TGHZHL7SPKXQXtxNPpNwq5ssyPLvHj9bpHq8eSqLHc7kWQhwlUefcbGBmkHndNWxk9hS0XaR7v8gw4cSLqPQMAd3m8kiI5Y0HcbMUKCqZUpoGnI+UaORdehcNZiP5hO96vVhHq9UjZzcKVNRNGqaBSLjz1gYYc5BF5HB2iPh31ph67bBq4Xw3ndsQ7vQI71TkwpqL7+0LL2XCzhglOQ1v3pMeCQhwEaUBU4+6YoJ7kd8JtDuKQ69KOOMQqw+NxNowiGcEAhQ4Ei2Kmu2exbtHrEUgPz16skTkGeYX7dzfpjVtjvPDtZ4G4w3qgTZ97QJaPHa3qBsdXwZptpzbY5jtWLkrOL4gyNbDDd8s9vxGhQpPXNfM+Bo3NcdfjYNEqzwSEAGrc0h+lO2LfqLmLUCq5uVJdf37u9z4DhNh2YmiMCz+yGj8Ro3pqkHnhOmgXr6+M2f7h3t61gz2bGRkV29F2lPdxi2HRCBZKu/3E3RmXMEGtbtaWaX13Z+5q7tfIz6X9hFQFPImUYM36EP4nj9tXhGtkHsRbQD0KuHoo6vkNOYKXZl5MKsjhJE0eKcbp+M6ahnhZNm+2JPUVsRzxuQq2lWCR0mNYTtCsqQWGVn7VSOXsM0eJOpS2k/7KlnVPFMhV+VjL5BKa/0DGRMz4zjl5PxjMU2MqOxAbo5gdZcHrR2nPAxkdD+4HfY6hHD8lY5cL8ll7jAFavYg2izY1MbZ3fBbq7vzj6v/MTjQgZvuJh8kbN3hDeC5KL4ySdXSfLnJgYtXlIHWQeTkjFYpBbfZlCJ0I5sm52UiWEsvTHuklCH92ZGC5z0jH4dmZtJJ4LF3rfT5ydJcBCULyYgaBB7gXlvwYrxV5VN8vB4Gc7A0tH0D5XhEv+3ZUUhcf0H7AlAmdxBGuHC6aC6xD7AeqUh1KKk4hXHxM9Kl6tk2C3hR8X9bWnw5LBJ2ZK8PxCTXrqNlzIDQY2+cSPEG/DzbVOGAe7CTTI9WqSi0ZbgCtMM1HS1QDpD8UwKWdEg7n+RF+PXwT9Ok3Oe3EALmvlcDEI72KYNyGDIzG1Oebd4Fh6kjo6fz6rmVFQCOxKTd/tb2Zft5mSc0GcHVsJGXl7XCJpQx17eXQS4wbm2Ry1YJ25CtbZ1rs8m0EILrD7H6NSM4rgAvmiE6C1uKcBn1eI8sBIS6ekFcoRB1dQnqGcYlB0RL6iunzNdE1BA9YHtZtGQNgcobgwjGqyo1yYrnaNmNZ5iKyyqZvk6H4TdqSDxOKTLbqeyMZbIyFqIgjW2ZTZTlLutPXn6sSUuGvVnSQEo7yyKQzOeyowsyyN66kPNd8xuD4/9WM18B9rO5Dqt9oH7V0PlJ4VKDrDN2/1b8j25JMGmN5n4L0wBJhdqFQkUtM5r0qvU7UqnXuSB7O67XHmhv6YyBLxzL34DnE2G12UBGNzAwYPCa0WASCUH3sLIpKU3PYCvmRcvz/20Uu98x1K1hjGFHOxQMQUBHj8XjMPGKCHfGGjpXEpeOvmqzxuvqg72vvDU9M+D4QcOxhmtmQsTGE5KNj2uo4XI3BFbxiEGLO2xhmXhtaWc30eD2ipUm3PwNaDg6nD+tpyCPFGTUl/W+VVvOF5R5DTzT8EyaqtWUFf55EEUszPAl14GL7X4p9FeC7NtIBPf3nsB/ah+O5wadUjPQIkOpE7bZ4EqBLyDWjMm0ZYrHf1SXOmT76I1vjgNBn4KOQgEWUA8L3qRH1SbHpNsV40JdHO7D9QgxmsOyK6GOKfbQyTrI9i0gQtQ7RsjOVn+PoPm1ojHVd1V8HnzLqm5CGrW8XsdK15nONKkNdsDqUlXmKXHb+AwJoMvu4EfyZ0FYnJQ+2IBaN7L3m2GNvPK2nAhNptgCaqKPOdjmTl6CzXWx7SfIOtn0AMQnNUv5zRHyjXwXQPP3wcNRIzCz4/O4v0gi3YvJ6GoE8oVnZ0pWb9a76ogadZVAUeUIbqDWY7ChXc4I3c9dwCLd/mSNcIdIq1Dl54ZG3sT4lpTwAvLdsnm9c12jjw5gzAK7z2zklw89ccGGHsDlqdTwl2HvmhanQpnlF9P9zz0nDTsmyxUZbImd1IoXLC5HXZsvR+Wc3u1bhXiR3AQUvDUnBQX6+RE0sSxCuDq5XLsnBL1d/1zyvrd2VtjlPmIHXgL0YRUftrdaWvcyvqlEUEV4o5T5qrb1PmSAGwLfDwh6whqB8NHiHJEcvWZ1jqnZHLQHPhDtBRXHZa37K66UvWddpu5+lmGYiZ1POxu25+dS2+cQoYudFbl0EZ1V2PGUU8bIPBzJFPyLBthsjcqaZvDVNe0bEAMp6xKyR+7gEm6pbIkDv97V5nXUnFDLF8Gf9llOALZplk8zqGPMMOmC2RmoI7IYmapKpWhkiKf9u4Bgkh5XKwCJ8MjB5Va8sL6o6Z0IGbf6qi8b5Y2kxaGqpQdVqjEtUNKDgKs31GQ4kG2BCeDdyPXwtbEB3rKkyw5Xd6iEl4CFgwJsm+Fdr4dbQhEeGUZcJ71t/5u/GUJTktlNO09S0Pb7EObGVKlHNXKQsN3ArPuI4pwe6rMNTuS8CaLm3Zgt5bnPwn3mK4qyJMbCeIzsDBHT4LrEcMzQ4h8ms+0Q1YjaPwgsnZ0TxT4PHnQsqyOqr3/fjpnfEnIl+jxB2nXxnqOeJa1gtKbEuWaVqV2Fj2wT/sdaXOoNzYtNdhsrFEjGV7+iA6ZRBwWuBzYa2qXshU0LtYH3Q8bTO2kCgyUYUn1YGtHLB5LNR+iiTJhdHkThGTtnU/pFuv8tdc2g8NUuufoRUT45/Vih4xHzzvjeU4lBuMgc7O0jKDXSBXlEnc8XHDtGJUCF/5mXGVR9mAH8HKhFzN/RK9kv5jtd2c4MxHwBC0W1Wc3Ie1t1DDnZBojNmsq1GZ4b0iPVHwZoC6CwSVN+QN+7dtCeUb0EOv7XUXSkd8NNIJTxtYE5D3xK+jvUL9aY3x7xdbzTE3wjcyyP8s7xRWwjz+KxlC9PuGVyi4YF9TCb04wJ6djKNIoFDs3m+fKYKoRZcFGZg3khnXPe3+YEm4Y2uIm93ti0B9otMY4bMt9fMK1pVj2aM1EXQh6Ef/TDabh7vyAm9NUwm1UI7sOd0bDcJWbfF5kr0h9TwUKtbfGRemjzgm1CcMrqf+0SuWjWnVjuTqvc1n+tNoQxxGftOlmS77LOPFDu0NFaXJtdwc2VJVzmxWpIVcb/eRMRSUMSTYZ7PBwVb6VxbUM1Uok3Mi6RIAAAAASUVORK5CYII=\n",

```
    "text/plain": [
     "<Figure size 864x432 with 2 Axes>"
    ]
   },
   "metadata": {
    "needs_background": "light"
   },
   "output_type": "display_data"
  }
 ],
 "source": [
  "acc = history.history['accuracy']\n",
  "val_acc = history.history['val_accuracy']\n",
  "\n",
  "loss = history.history['loss']\n",
  "val_loss = history.history['val_loss']\n",
  "\n",
  "epochs_range = range(100)\n",
  "\n",
  "plt.figure(figsize=(12, 6))\n",
  "plt.subplot(1, 2, 1)\n",
  "plt.plot(epochs_range, acc, label='Training Accuracy')\n",
  "plt.plot(epochs_range, val_acc, label='Validation Accuracy')\n",
  "plt.legend(loc='lower right')\n",
  "plt.title('Training and Validation Accuracy')\n",
  "\n",
  "plt.subplot(1, 2, 2)\n",
  "plt.plot(epochs_range, loss, label='Training Loss')\n",
  "plt.plot(epochs_range, val_loss, label='Validation Loss')\n",
  "plt.legend(loc='upper right')\n",
  "plt.title('Training and Validation Loss')\n",
  "plt.show()"
 ]
},
{
 "cell_type": "code",
 "execution_count": 26,
 "id": "0f4a6d6b",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:53:55.277899Z",
   "iopub.status.busy": "2021-12-21T16:53:55.277140Z",
```

```
   "iopub.status.idle": "2021-12-21T16:53:55.684759Z",
   "shell.execute_reply": "2021-12-21T16:53:55.684183Z",
   "shell.execute_reply.started": "2021-12-21T16:26:20.769667Z"
  },
  "papermill": {
   "duration": 1.110483,
   "end_time": "2021-12-21T16:53:55.684923",
   "exception": false,
   "start_time": "2021-12-21T16:53:54.574440",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "array([[0.07391414],\n",
     "       [0.02940544],\n",
     "       [0.07304674],\n",
     "       ...,\n",
     "       [0.01020581],\n",
     "       [0.00300625],\n",
     "       [0.00155538]], dtype=float32)"
    ]
   },
   "execution_count": 26,
   "metadata": {},
   "output_type": "execute_result"
  }
 ],
 "source": [
  "model.predict(X_test, batch_size=32)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 27,
 "id": "5d5f6649",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:53:57.097640Z",
```

```
   "iopub.status.busy": "2021-12-21T16:53:57.096944Z",
   "iopub.status.idle": "2021-12-21T16:53:57.215562Z",
   "shell.execute_reply": "2021-12-21T16:53:57.216119Z",
   "shell.execute_reply.started": "2021-12-21T16:47:33.831667Z"
  },
  "papermill": {
   "duration": 0.832166,
   "end_time": "2021-12-21T16:53:57.216334",
   "exception": false,
   "start_time": "2021-12-21T16:53:56.384168",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "Y_pred = model.predict(X_test)\n",
  "y_pred=[]\n",
  "for x in Y_pred:\n",
  "    if x>.5:\n",
  "        y_pred.append(1)\n",
  "    else:\n",
  "        y_pred.append(0)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 28,
 "id": "f9190dd3",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:53:58.642576Z",
   "iopub.status.busy": "2021-12-21T16:53:58.632649Z",
   "iopub.status.idle": "2021-12-21T16:53:58.645317Z",
   "shell.execute_reply": "2021-12-21T16:53:58.645838Z",
   "shell.execute_reply.started": "2021-12-21T16:47:40.933940Z"
  },
  "papermill": {
   "duration": 0.728792,
   "end_time": "2021-12-21T16:53:58.646011",
   "exception": false,
   "start_time": "2021-12-21T16:53:57.917219",
```

```
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "data": {
   "text/plain": [
    "[0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 1,\n",
    " 0,\n",
    " 1,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 1,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 1,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
    " 0,\n",
```

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",

" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",

" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",

```
      " ...]"
     ]
    },
    "execution_count": 28,
    "metadata": {},
    "output_type": "execute_result"
   }
  ],
  "source": [
   "y_pred"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 29,
  "id": "da9618f1",
  "metadata": {
   "execution": {
    "iopub.execute_input": "2021-12-21T16:54:00.150409Z",
    "iopub.status.busy": "2021-12-21T16:54:00.149707Z",
    "iopub.status.idle": "2021-12-21T16:54:00.167312Z",
    "shell.execute_reply": "2021-12-21T16:54:00.167823Z",
    "shell.execute_reply.started": "2021-12-21T16:49:25.309418Z"
   },
   "papermill": {
    "duration": 0.815276,
    "end_time": "2021-12-21T16:54:00.167994",
    "exception": false,
    "start_time": "2021-12-21T16:53:59.352718",
    "status": "completed"
   },
   "tags": []
  },
  "outputs": [
   {
    "data": {
     "text/plain": [
      "[0,\n",
      " 0,\n",
      " 0,\n",
      " 1,\n",
      " 0,\n",
```

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 1,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",

" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",

" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 1,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",

" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 1,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 1,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",
" 0,\n",

```
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 1,\n",
      " 0,\n",
      " 1,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 1,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 1,\n",
      " 1,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " 0,\n",
      " ...]"
     ]
    },
    "execution_count": 29,
    "metadata": {},
    "output_type": "execute_result"
   }
  ],
  "source": [
   "y_test=list(y_test)\n",
   "y_test"
  ]
 },
```

```
{
 "cell_type": "code",
 "execution_count": 30,
 "id": "9ad5589f",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:54:01.602720Z",
   "iopub.status.busy": "2021-12-21T16:54:01.600459Z",
   "iopub.status.idle": "2021-12-21T16:54:01.609740Z",
   "shell.execute_reply": "2021-12-21T16:54:01.609059Z",
   "shell.execute_reply.started": "2021-12-21T16:49:39.348560Z"
  },
  "papermill": {
   "duration": 0.737052,
   "end_time": "2021-12-21T16:54:01.609885",
   "exception": false,
   "start_time": "2021-12-21T16:54:00.872833",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "from sklearn.metrics import confusion_matrix\n",
  "cm = confusion_matrix(y_test, y_pred)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 31,
 "id": "628fbfcd",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:54:03.041018Z",
   "iopub.status.busy": "2021-12-21T16:54:03.039869Z",
   "iopub.status.idle": "2021-12-21T16:54:03.046765Z",
   "shell.execute_reply": "2021-12-21T16:54:03.047307Z",
   "shell.execute_reply.started": "2021-12-21T16:49:40.750783Z"
  },
  "papermill": {
   "duration": 0.725383,
   "end_time": "2021-12-21T16:54:03.047486",
```

```
   "exception": false,
   "start_time": "2021-12-21T16:54:02.322103",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "array([[1538,   54],\n",
     "       [ 211,  197]])"
    ]
   },
   "execution_count": 31,
   "metadata": {},
   "output_type": "execute_result"
  }
 ],
 "source": [
  "cm"
 ]
},
{
 "cell_type": "code",
 "execution_count": 32,
 "id": "f2438e23",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2021-12-21T16:54:04.488522Z",
   "iopub.status.busy": "2021-12-21T16:54:04.487800Z",
   "iopub.status.idle": "2021-12-21T16:54:04.723666Z",
   "shell.execute_reply": "2021-12-21T16:54:04.724131Z",
   "shell.execute_reply.started": "2021-12-21T16:49:42.548327Z"
  },
  "papermill": {
   "duration": 0.948203,
   "end_time": "2021-12-21T16:54:04.724369",
   "exception": false,
   "start_time": "2021-12-21T16:54:03.776166",
   "status": "completed"
  },
```

"tags": []
},
"outputs": [
{
"data": {
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAATkAAAE9CAYAAABwcBXnAAAAOXRFWHRTb2Z0d2FyZQB
NYXRwbG90bGliIHZlcnNpb24zLjUuMCwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy8/fFQqAAAACXB
IWXMAAAsTAAALEwEAmpwYAAAS40lEQVR4nO3debhd872A8fffrRCKIUqEqhhgyiBAkgmjV1FTU
XCqh7U21FG1pFeUaUjQ1lHv10tuE8lBKzUOCGtJeIYaIqHloEA0JMmicpC1Jzu/+cXbSk9Eue521z6v/v53n
Ok73XWmev7+Y877PWHiOlhCTlaqWyB5CkIhk5SVkzcpKyZuQkZc3IScqakZOUtXZlD9BStOuYon2nss
dQndp2i43KHkF16o03Wyx1Irr6ilz7TnTo8dWyx1CdGvf4pWWPo0Dq18w79lrvO01VJWTNykrJm5CR
lzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNy
krJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsG
TlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUta
MnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5S1dmU
P8O9kxLDDGbRLb6bPaqTfIT8D4LTv7M0RBw1g+ntzABh26Z3c+/AL9NtyYy49YwgAET8xN3c+cdn
APj+4bsx9MABpJR4ftJUjp2LR98OL+cO6VW0WPzrnRavRMNDQ20a9eOcY9PWLTu4v++iFNPPpEp0
6bTuXXPnEqesT4VGLiL2An4BNAC/TimdV+T+T+6t01ox5jxA0P8utzvrHY8kuu/SMXXznNmsWXPvzqVnQ+/
gAULmliv8xo3DX2OT7z6U4cO+QLbPPuV4fzjg3lce/4f4RHPKlvvlw76vHWvCsqwe8f+ONSEZsyZQpj7
r+PDTfaqKSp6l9hp6sR0QD8EhgE9AKGREfvfGSvEnzYH9AKGRESvovbXFoyb+CqzZv+tqm3//o95LFjQBECH9iuTUlq0rl1DAx
07rExDw/p0XKU6U906bPLmRe1b+TT/whw8+9gIoe5S6VeRjcv2BSSml11JKHwGD2H8D
acyYtjhrNmp46Ll2/femCdvPo0JN/0nxw3/HQsWNDF1+mwu/s0Yx/k7qYTp1Roign0H
DWRA/75ccfllAIy68w7WX78LW/fpU/J09a3IyHUBprS/mZlmVq4+8hePN9zjvhoEXrn
njuDfoePJzPfe0CTjpiB3at2PNTh3ZZ9et2GKfYWw68DRW69iewXtvX+I9UGsY838P8+gTE7l99D2M/N
UvefrFihsVxw3s848ydnlz1a3Sv92dWIOCoiJkTEhEPD/ZPt
UvefihsVxw3s848ydnlz1a3Sv92dWIOCoiJkTEhDT/72WP0+rendVIU1MipcSVt46jX++Nl9rm5dffPm
DLzddn9x16MnnqTGa8N4f585u47rcXmA0ITa27uk/3OwhXAi4vX3fFiIZxTEh1Z4TbTtGnTrspA
9+SY79d+Ot99+u+RJ60+RkXsL2L2LDF9Q0qyxFA+LK5TaydvYRDAQMRmJF81fpk7fsNHI8/LjWn4Z7LyrWyvc
TUND8/+ejT67Fj02WY83ps5kytuz6L/VJnRcZWUAduvfg5dff6f1B1ermTt3Lo2NjYsuP+P3D/ffTttz1/mfouL
0+azMuTJtNlgw14dPxE1ltvvvZKnrT9FPrv6BNAtIjahOW6DgcMK3F/du/rcoXy+bzc6r7k6k+69nZ
nZj6x4bkFLijWmz+P5PrwdgwLabcuI3xMp9nbBzgKamhLH/+wgFhrOpIzEeKl+7MfMXNPH0Rmna
3jSr5nKtK777zDoQcfCMD8BfM5dPBhDPzSXiVXZ9ftan7jEXsDF9P8EpKpFsDP8EpIrU0rDV7T9Squumzr0+G
ph86hte++JS8seQXVq5x368eSTE5b5FHOhr5NLKd0N3F3kPkp0UKpmTkJGXNyEnKmpGTlDUjJ
ylrRk5S1oycpKwZOUlZM3KSsmbkJGXNyEnKmpGTlDUjJylrRk5S1oycpKwZOUlZM3KSsmbkJGXNyE
nKmpGTlDUjJylrRk5S1oycpKwZOUlZM3KSsmbkJGXNyEnKmpGTlDUjJylrRk5S1oycpKwZOUlZM3K
SsmbkJGXNyEnKmpGTlDUjJylrRk5S1oycpKwZOUlZM3KSstZueSsiohFIC69W/k2VyymltEbBs0nSJ7bc
yKWUOrXmXyJhJUhKpOVyPicxHxzcrlzhGxSbFjSVJtfGTkImIY8GPB1+wFy
AlNJUwFNZSW1CNZH7MKWUWFjSVJtfGTkImIY8GPg1Mqi9sC1RQ4lSbVZHcgcB+wFy
AlNJUwFNZSW1CNZH7MKWUWqDwJERGrFTuSJNaGrFTuSJNVONZG7MSJGAmtGxJHAA8DlxY4lSbWx4lSbWx3GdXF0
opXRgRXwTeB7o7kZMqgz7moDZ6aU7i98LkmqQ0MU5uz4ohsd3h6aUaYFoPKpNbSqGv6BtBTC4Haeky0oQm0KcXuYc
SaqFao7kTgK2nSBIIitYFHgCuLHEySaqGaJx5mAo0trdWlkS3V0NP5mAoNBqz
bJH1iKzpdXfiC31crPwdUdw4klRbK3qPwPzQ4vOVsfWelM99Vw4lRbwkUk4AjAqwOvVmHwheqGaPoqQPl7MBwOtFC
4CVgE+AsYDLwRIEbiKzpdXfiC31crPwdUdw4klRbK3qPwPzQ4vOVsfWelM99Vw4klRbwkUk4AjAqwOvVmHwheqGaJx5mAo0trdWlkS3V0NP5mAoNBqz
4CVgE+AsYDLwRIEzSVLNVBO5tVNKVwDzdUkoPppSOAOyeqF9pTVK0Q5n0P6rBuqn0YdVdxUjPFk4Kk9QmVPM6uXmSHSW4D4kBFBcB6JSjIikeCnfdXF

mqnmsj9NCI+BfwIuARYA/hhoVNJUo1U8wb90ZWLs4Hdih1HkmprRS8GvoR/fpHNUlJKxxUykSTV0Iq
O5Ca02hQVvbtvyOgxF7X2btVGzGz8oOwRVKfmNy33eGyFLwa+upBpJKkV+eXSkrJm5CRlzchJylo1nw
zcPSLGRMRzletbR8TpxY8mSZ9cNUdyl9P8xdLzAFJKzwCDixxKkmqlmsitmlIav8Sy+UUMI0m1Vk3kZ
kTEZvzzy6UPBqYVOpUk1Ug17139LnAZ0DMi3gJeB75W6FSSVCPVvHf1NWDPiFgNWCml1PhRvyNJ
9aKaTwY+c4nrAKSUzi5oJkmqmWpOV+e2uLwKsA/wYjHjSFJtVXO6utg75iPiQuDewiaSpBr6OO94WB
XYoNaDSFIRqnlM7ln++blyDcA6gI/HSWoTqnlMbp8Wl+cD76SUfDGwpDZhhZGLiAbg3pRSz1aaR5Jqa
oWPyaWUFgAvR8RGrTSPJNVUNaerawHPR8R4WrycJKW0X2FTSVKNVBO5MwqfQpIKUk3k9k4p/bjl
gog4H3iwmJEkqXaqeZ3cF5exbFCtB5GkIqzoe1ePAY4FNo2IZ1qs6gSMK3owSaqFFZ2uXgfcA5wLnNJie
WNKaVahU0lSjazoe1dnA7OBIa03jiTVlt/WJSlrRk5S1oycpKwZOUlZM3KSsmbkJGXNyEnKmpGTlDUjJ
ylrRk5S1oycpKwZOUlZM3KSsmbkJGXNyEnKmpGTlDUjJylrRk5S1oycpKwZOUlZM3KSsmbkJGXNyE
nKmpGTlDUjJylrRk5S1oycpKwZOUlZM3KSsmbkJGXNyEnKmpGTlDUjJylrRk5S1oycpKwZOUlZM3K
SsmbkSjL1rSkcuv9A9thpG/YcsC1XjrwUgLvuuIU9B2xL184deeapJxdt/96smRy6/0C22Ghtzjj5ByVNrdbyo
+8dRZ/uG7LHgO0WLXvhuWfYb+AX2GPnvgwdchCN778PwK03Xc/AXfov+tlw7Y48/+zTZY1edwqLX
ERcGRHvRsRzRe2jLWtoaMfpZ5/PmEf/xO33juU3V4zglZdepHvPLRl59Q3sMOBzi23focMqnHjqME476
7ySJlZrOuSwr3PtTXcutuyk44/h1GHnMGbck+z15f0Yccl/AXDQIUO4b+x47hs7nl+MuJKNNu7Kllv1KWP
sulTkkdxVwF4F3n6b9pn1PstWfbYFYPVOndi8W0/emfYW3Xr0ZLNu3ZfaftXVVmP7HXemQ4cOrT2qSr
DjgM+z5lprLbbstUl/ZscBnwdgl1334O5Rty/1e3fccgP7HXRIa4zYZhQWuZTSWGBWUbefkyl/mczzz/6Jbf
r2L3sU1bHuPXtx792jABh9x61MnfrmUtuMuu1m9j/o0NYera75mFzJ5s6Zw9FDh3Dm8AvptMY8AvptMYaZY+jOn
bRJSP5zRUjGbTbTsyZ08jKK7dfbP3ECeNZpeOq9Oy1ZUkT1qd2ZQ8QEUcBRwF02WDDkqdpXfPmzeP
ooYM54ODBDNr3gLLHUZ3bvHsPrrv1LqD51HXM/b9fbP2dt97EAV/5ahmj1bXSj+RSSpellPqllPp9eu11y
h6n1aSUOPm477B5954ceezxZY+jNmDG9HcBaGpq4hcXncvXh3570bqmpiZGLj8ctQ+lHcv+uJjz+C
LfeeB09e/Vm0BeaH4s76fSz+fCDDxh2ygnMmjmdwbw45kF69t5W3l1eRSWGvT72WlsbGTevA+57+5RXHPza
Lr33KLMu6GCfPfbX+fRcQ8xa+YM+m25GT865XTmzp3L1VeMAGDQpDQPgdw6OH/sWj7xx55iPXX34CN
u25a1sh1K1JKxdxwxPXArkBn4B1gWErpihX9ztbb9E2j//BIIfOo7WuIsidQvdp79wE8/dSTy/Lq
pKjblqRqlf6YnCQVychychJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUt
aMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iyc
pa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypq
Rk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5CRlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWTNykrJm5C
RlzchJypqRk5Q1Iycpa0ZOUtaMnKSsGTlJWYuUUtkzLBIR04E3yp6jjnQGZpQ9hOqSfxuL2ziltM6yVtRV
5LS4iJiQUupX9hyqP/5tVM/TVUlZM3KSsmbk6ttlZQ+guuXfRpV8TE5S1jSk5Q1I1eHImKviHg5IiZFxC
llz6P6ERFXRsS7EfFc2bO0FUauzkREA/BLYBDQCxgSEb3KnUp15GpgYNlDtCVGrv7sDExKKb2aUpoP3AQcXPJMqhMppQeBGWXP0ZYYufrTBXiz2e23KuskfQIYOUlZM3L15y1g02a3NwHeLmkWqc0zcvXnSaB7RGwWER2BrwF3ljyTVLeMXJ1JKS0EvgeMBl4Abk4pPV/uVKoXEXEjMA7oERFvRcRRZc9Ub74YWJJkvN4VelzZBMCA3nyRjaUjDQrF0bz+7lqX4yORr/NLaqcGfDR1QtYi
i4hNIqI9MBi4s+SZpDbLyNWDHXIvtHBJ84HvLnUrefLnUr1IiKuBJEmxHxrbJnqne+40FS1jySk5
Q1Iycpa0ZOUtaMnKSsGTlJWTNyKlxE7BoBoRoyuX91vRJ6ExJylXsc1vEJoRcezH2MKOyGkHZPIuLEapcvcvsc1VEXHwv7Cvrn
4KSNth5PSxVT4x5V+SUrozpXTeCjZZE/iXIyctj5HTUiVVq2smxwR50NfiYCB7W47j4sWIuDkiVq2smxwR50fEROCQiBgQEeQ
YEY9GxMSIuCkiVq9st1flNiYCB7W47aERcWnl8+bgSCLlPEWlyMwA4D9gsIv4UET+bvHdSDRwDRwREc9ExF
ktbuu0iHglIh4GelRv46s3M7TEXHLwvtUsWdETKjc/Rsi4uct9v2dT/rfVq3PyFXmk9K6Y+ucThhRzqZ/aliOCpeLdu2ua4CU2p47MwD3Rgvxt8RVm1ce1rKy
uZqaUtgMeAE4HrcJGrAJcD+wJ9gfWS4BrgXA04BXkuEBK0obZNSOikiBgLdaP4Iq
m2AvhGxS0T0pfktb9sAewPbV3Gfbk0pbV/Z34tAy3cLdK3s48vAiMp9+BYwO6W0feX2j4yITarYj+pIu7I
HUN2aklIaV7l8LXAccGHl+g2Vf3ek+YM9x0UEQHua33LUE3g9pfQRgFr4OFjhjqGfvYHfgGQEppATA7ItZ
aYpuBlZ+nKtdXpzl6nYDbUkp/q+yjmvf39o6In9J8Srw6zW+dW+jGlFIT8OeIeK1yHwYAfZKWk1yHwYCW7d4vO5TlX2/
UsW+VCeMnJZnyff7tbw+t/JvAPenllIa03DAitqnm1AjitqnhHAGcm1IaucQ+fvAxbusbWtcW6Zd3fA

L6fUmoZQyKi68fYt0ri6aqWZ6OI2Kly+TDg4WVs8xiwc0RsDhARq0VEd+AloGtEbFbZbsgyfhdgDHBM
5XcbIuJTQCPNR2kL3Qsc0eKxvi4RsS4wFjggIjpGRCeaT40/SidgWkSsDBy+xLpDImKlysybAi9X9n1MZ
XsiontErFbFflRHjJyW52XguxHxIrAW8KslN0gpTQeGAtdHxDNUTlVTSv+g+fT0rsoTD+8uZx/HA7tFxL
PAk0CvlNJMmk9/n4uIn6eU7gOuAx6tbHcz0CmlNJHm0+angXto/oiqj3IG8DgwjuYQt/QXYHzlto6u3IdfA
y8AEysvGRmJZz9tjp9CoqVUTsdGp5R6lz2L9El5JCcpax7JScqaR3KSsmbkJGXNyEnKmpGTlDUjJylrRk
5S1v4fqFlJ4TT5IccAAAAASUVORK5CYII=\n",
      "text/plain": [
       "<Figure size 360x360 with 1 Axes>"
      ]
     },
     "metadata": {
      "needs_background": "light"
     },
     "output_type": "display_data"
    }
   ],
   "source": [
    "from mlxtend.plotting import plot_confusion_matrix\n",
    "fig, ax = plot_confusion_matrix(conf_mat=cm , figsize=(5, 5))\n",
    "plt.show()"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": null,
   "id": "4b9976f9",
   "metadata": {
    "papermill": {
     "duration": 0.704178,
     "end_time": "2021-12-21T16:54:06.131840",
     "exception": false,
     "start_time": "2021-12-21T16:54:05.427662",
     "status": "completed"
    },
    "tags": []
   },
   "outputs": [],
   "source": []
  }
 ],
 "metadata": {
  "kernelspec": {

```
   "display_name": "Python 3",
   "language": "python",
   "name": "python3"
  },
  "language_info": {
   "codemirror_mode": {
    "name": "ipython",
    "version": 3
   },
   "file_extension": ".py",
   "mimetype": "text/x-python",
   "name": "python",
   "nbconvert_exporter": "python",
   "pygments_lexer": "ipython3",
   "version": "3.7.12"
  },
  "papermill": {
   "default_parameters": {},
   "duration": 180.885587,
   "end_time": "2021-12-21T16:54:09.492708",
   "environment_variables": {},
   "exception": null,
   "input_path": "__notebook__.ipynb",
   "output_path": "__notebook__.ipynb",
   "parameters": {},
   "start_time": "2021-12-21T16:51:08.607121",
   "version": "2.3.3"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 5
}
```