```
{
 "cells": [
  {
   "cell_type": "markdown",
   "id": "9f79b0f6",
   "metadata": {},
   "source": [
    "# Time Series Analysis"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "914518cc",
   "metadata": {},
   "source": [
    "## 1. Plot the data with proper labeling and make some observations on the graph.\n"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 1,
   "id": "0b8cc649",
   "metadata": {},
   "outputs": [],
   "source": [
```

```
  "# import libraries \n",
  "import pandas as pd\n",
  "import matplotlib.pyplot as plt\n",
  "import numpy as np\n"
 ]
},
{
 "cell_type": "code",
 "execution_count": 2,
 "id": "f90ce069",
 "metadata": {},
 "outputs": [],
 "source": [
  "# Load data\n",
  "data = pd.read_csv('us_retail_sales.csv')"
 ]
},
{
 "cell_type": "code",
 "execution_count": 3,
 "id": "d73cee4e",
 "metadata": {},
 "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>YEAR</th>\n",
 "      <th>JAN</th>\n",
 "      <th>FEB</th>\n",
 "      <th>MAR</th>\n",
 "      <th>APR</th>\n",
 "      <th>MAY</th>\n",
```

```
"     <th>JUN</th>\n",
"     <th>JUL</th>\n",
"     <th>AUG</th>\n",
"     <th>SEP</th>\n",
"     <th>OCT</th>\n",
"     <th>NOV</th>\n",
"     <th>DEC</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"     <th>0</th>\n",
"     <td>1992</td>\n",
"     <td>146925</td>\n",
"     <td>147223</td>\n",
"     <td>146805</td>\n",
"     <td>148032</td>\n",
"     <td>149010</td>\n",
"     <td>149800</td>\n",
"     <td>150761.0</td>\n",
"     <td>151067.0</td>\n",
"     <td>152588.0</td>\n",
"     <td>153521.0</td>\n",
"     <td>153583.0</td>\n",
"     <td>155614.0</td>\n",
```

```
"   </tr>\n",
"   <tr>\n",
"    <th>1</th>\n",
"    <td>1993</td>\n",
"    <td>157555</td>\n",
"    <td>156266</td>\n",
"    <td>154752</td>\n",
"    <td>158979</td>\n",
"    <td>160605</td>\n",
"    <td>160127</td>\n",
"    <td>162816.0</td>\n",
"    <td>162506.0</td>\n",
"    <td>163258.0</td>\n",
"    <td>164685.0</td>\n",
"    <td>166594.0</td>\n",
"    <td>168161.0</td>\n",
"   </tr>\n",
"   <tr>\n",
"    <th>2</th>\n",
"    <td>1994</td>\n",
"    <td>167518</td>\n",
"    <td>169649</td>\n",
"    <td>172766</td>\n",
"    <td>173106</td>\n",
"    <td>172329</td>\n",
```

```
"    <td>174241</td>\n",
"    <td>174781.0</td>\n",
"    <td>177295.0</td>\n",
"    <td>178787.0</td>\n",
"    <td>180561.0</td>\n",
"    <td>180703.0</td>\n",
"    <td>181524.0</td>\n",
"   </tr>\n",
"   <tr>\n",
"    <th>3</th>\n",
"    <td>1995</td>\n",
"    <td>182413</td>\n",
"    <td>179488</td>\n",
"    <td>181013</td>\n",
"    <td>181686</td>\n",
"    <td>183536</td>\n",
"    <td>186081</td>\n",
"    <td>185431.0</td>\n",
"    <td>186806.0</td>\n",
"    <td>187366.0</td>\n",
"    <td>186565.0</td>\n",
"    <td>189055.0</td>\n",
"    <td>190774.0</td>\n",
"   </tr>\n",
"   <tr>\n",
```

```
"     <th>4</th>\n",
"     <td>1996</td>\n",
"     <td>189135</td>\n",
"     <td>192266</td>\n",
"     <td>194029</td>\n",
"     <td>194744</td>\n",
"     <td>196205</td>\n",
"     <td>196136</td>\n",
"     <td>196187.0</td>\n",
"     <td>196218.0</td>\n",
"     <td>198859.0</td>\n",
"     <td>200509.0</td>\n",
"     <td>200174.0</td>\n",
"     <td>201284.0</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>"
],
"text/plain": [
"   YEAR    JAN    FEB    MAR    APR    MAY    JUN     JUL     AUG \\\n",
"0  1992  146925  147223  146805  148032  149010  149800  150761.0  151067.0  \n",
"1  1993  157555  156266  154752  158979  160605  160127  162816.0  162506.0  \n",
"2  1994  167518  169649  172766  173106  172329  174241  174781.0  177295.0  \n",
"3  1995  182413  179488  181013  181686  183536  186081  185431.0  186806.0  \n",
```

```
    "4  1996  189135  192266  194029  194744  196205  196136  196187.0  196218.0   \n",
    "\n",
    "       SEP       OCT       NOV       DEC  \n",
    "0  152588.0  153521.0  153583.0  155614.0  \n",
    "1  163258.0  164685.0  166594.0  168161.0  \n",
    "2  178787.0  180561.0  180703.0  181524.0  \n",
    "3  187366.0  186565.0  189055.0  190774.0  \n",
    "4  198859.0  200509.0  200174.0  201284.0  "
   ]
  },
  "execution_count": 3,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "# display data\n",
 "data.head()"
]
},
{
 "cell_type": "code",
 "execution_count": 4,
 "id": "1e09688b",
 "metadata": {},
```

"outputs": [],
"source": [
"# data preprocessing \n",
"\n",
"# Melt the data into a long format\n",
"data_processed = data.melt(id_vars=['YEAR'], var_name='MONTH', value_name='SALES')\n",
"\n",
"# Convert YEAR and MONTH into a datetime object\n",
"data_processed['DATE'] = pd.to_datetime(data_processed['YEAR'].astype(str) + data_processed['MONTH'], format='%Y%b')\n",
"\n",
"# Sort by date\n",
"data_processed = data_processed.sort_values('DATE')\n"
]
},
{
"cell_type": "code",
"execution_count": 5,
"id": "2b20784e",
"metadata": {},
"outputs": [
{
"data": {
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAA8sAAAHjCAYAAAAUkY3dAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90
bGliIHZlcnNpb24zLjQuMywgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/MnkTPAAAACXBIWXMAAAsTAAALE
wEAmpwYAAC0UUlEQVR4nOzddXyV5f/H8dd9ekUMNrpBUkIGiAJTUUJAwsAO7AATRcDAAhWMn/m1FRO

RMAgVBUSkpQREkc7BqMXp+/fHYWcbY7AB6/fz8fDh2X3uc1/Xfabb3udzhWGapomIiIiIiIiIhFmKugMiIiIiIiIix
Y3CsoiIiIiIiMhRFJZFREREREREEjqKwLCIiIiIiInIUhWUREREREREGRoygsi4iIiIiIiBzFVtQdEBGR4m3YsGFMnjwZgE
8++YQOHTpke37btm107doVgPbt2zN+/Phsz69fv54PP/yQhQsXsnfvXiwWC1WrVqVjx44MGjSIWrVqnbAPC
xcu5Prrrz/mczabjjZiYGOrVq8eAAQO47LLLLMAzjZG4VgGAwyDfffEOHDh2oXbv2SV2jcePGADRp0oSpU6S
GnSJB599FEAHn30UW688cYTXud0vHe5ue6661i0aB
Eaixcvply5cid9rdNh7dq1fPHFFyxcuJA9e/Zgmibx8f/Zgmibx8fG
0b9+ea6+9liZNmhRp/47neP99HsusWbOYPHkyr7/+OgBvvPEGF154YUF154YUF1T0RETpLCsoiIFJiffvqJ++67D7/fn+
34xo0b2bhxI1OmTOHdd98lISHhpNvw+/3s2zZMv777z8eeeSRk7rW6tWreeeKJJ1i9ejVTpkw56
DoXx3hUXH3zwAS+++CLBYDDDb8c2bN7N582YmTpzIHXfsn+Xu/Xe5WEfdv99KyiIgkh8KyiIgUiNTUVIYPH47fpdD
Jw4EDOPPNMPB4P33//PQsWLCAtLY1HHnmEn376CYslbzODOnToEK7KFfPHFFwB89NFHXHXHP
NNdSsWTPfff70009ZvXp1vl93tDfeeAOA6Ojok3p9Qb13xdE888/D4BhGFx88cV07twZm83G0qVLmT
hxIj6fj7feeguHw8Fdd91VxD3OqVGjRuHveHveYa7774bgNjYWJ5++ulsz1WqVIImLL76Ypk2bAtCyZcvC6aiIiOSLw
rKIiBSIJUUuWcOjQIQBuuukm7r///Bz/fvz3p2/fvz7779s27aNdevW0axaFMmRi/wLfYSfr+Ub
gwGmT9/PldcccXpu5l8l8OtUhtQX13hU3+/fvZ/To0eGvX3rpS6++OLw13m1+eEeNnOHv9d/fr06XN/9dfr06XN
Kw88LQmxsbK7f84iIiGM+16BBAxo0aFDQXRMRkVNQcj+KFhGRYi0tLS38ODMzM9tzvXr14vXXX+fll19m
kqVKp1ye40aNo/ztoWwJo1a7jzzjtp164tp164dLVu2pG/fvnz88ccEAoRatweDwD++eD++88
w59+vShbdu2tGzZkgsuuIChQ4eyYa5vwbu0xTdaxZjdT6lZzkm80/HGSgd1Vdq8xZzRgd2Nly49N+Wc/bwTBC8tcJBiSChOk8SNd9NgwRw+uHDjTJ0c4eh2/8d3sJLqY1Awr3O6bAOdJ4mj18ApwRdvnrg47gnsI5LTW4bHCRHNdNz7ohoAiUiONKjwQOZ8qiFi
9rpGhfr6Kvt+AVaFMDTnbSxvcsSrIrD5iZcbJpdN4XfPdaS4p5NTPdDTh5oeX9oTmY0LT41AzeH1ASpx9O0OBXeEjBqAGhmzu57vepR2ZUkFIRvOjalKWHGhqmI8xNpVpIbfabjYmNaIS5zrRxffmpWQzatsMgfS/uvyF3UzVCVAl+R0
aA5Ec+U/2MD3XsZUv+33KhKbHgrn37x9kQMzhwUczW2duPyspanL3rHBOYmdxx3dfGa1e7Xx4UafZo1Uq4+Py9ZkvbTfdtq4X1P9f7ph4YusTvRPfTdTMxEUBtvCKaeLrX/EagB6GbIutnpZ76JWvGgZM0O9ye9ENNgvYqmrrTUn
n3B5mPHzUdRYmTXX5vLvV/AzbN3/AJ+72TvOaR6RtKDLo9TWvu6wBvHJGcDjbWFsh2vWKjW7mz6w7fw4U4r/r3GlA5/GfrkzgCN+tBQ8jsvz9TSh94pXrJZtJBqaGEJ8Lpf/eCfJzR
ezz4zG6McEtfYnNMfjvytWKMC6Lt43U6Vpu7y+Y6T9/3+hzH/dhvp9JTqnSTW2Y/0f5KA0fb/AqE3R3IHsjtzvU5SkRQEfk51fKeL8Ep3z+l
pz52GeXT4d/ZdaUeyJAgt3vqadf1Y4YkqgHPDe/DvoxOajk73T9+z/+oPW/bsp0PHnEVZYnvlD1TyWJ7R1dT/KjAyk2Jmbf1O8+rsl7DXAYj+/
bWlnGT1mb3N4ejs7M9xyJ2+SzQRE9PN8rP3ty4nUv9sAiXMk7lwEAYwJp4FH0nAfOsdfiMP/uydhvFxzGA0fS78/EqIrsmbO5o5/IEKwD3/U
YpQGOlwdbyXP2tMNTw1Vtzt5p7GR+Ha9Ga6vCiYHYLKH0U0kZqQLUJyyfsDsgv2mnFDq3VeWZV6sKzxgAndFmdmFjiMNQjD0cMc5m8hnLvb5H
dJ7TDdVwWdjebXsL+HX5rfDX1fGHUPvGbd/Ol3BxP1zVsFc7LmiHD2ovbvnKHs+MKFAt7IZSSnZx+V/TQNcL9vnXTp3o/N+/Ns4o3VNuHkz3T
DiPe6yo+cYy8x9X54Hw6+0hxQaJ0c/IVxmqdKoyZ5lgMWHnFDzYomIA+YNaDz5kJIWvJ2tnMA1/ZGcfy9Q+7+QmOiVw4OT0b/8flO6n0Hqi1Owi
+K2FetGBDxcbnrDMUAKRFr6Q2pXS7+uajJVKgt+yJGFVzvqYo0Rq6dH1Ri9wpPCyPJU8+4wPbGvymtNvNAbMqsrhAJ/7JFmwRFQxcE6ab9Mbwp+
x0yuMO8MGn1l8+Y9OAZW9Kb4mMoVyAxOGeAFZhbUbVsmLqrWjiJX6K+Vit2mG/Rv/QRUuL2AnJlP5cxLMeEIAfsg7ICvL4+Jw/eKGX4Ytoc+pG
4NAngNxO5GPPzJDsGMgPiiJMGxOPzWlMrZoDcdHtjdJ26V1I3MhH5dkRcBhe2Bb5jCXm4jtQ4zOF5L4D3RlLgvoHhzCwh3z1UCEyRVZJJcWZn/
OMHc7aWa1fWX/+OmVYvX6QCVBOHO5O0zuRwG7kDa08wZbvxjYgUtA6YV2ohGLWrjuZPzbWS6LGuoCTWXPMZS+zM/yC4aNWK1PUuQvKzkNTOXjt3p

vffw+/3cdtttVKhQgYSEBNq1a8cFF1yQpz2qs4bgrPO0j5Y1mGddYXzAgAE899xzHDhwgHnz5nH++eeHh2BX
rFiR8847L8d9vfrqq7z66qs52tixYwd79uwhPj4+2/HTOVUgP6pVqxYOyhBaQTsjLLdv3z58PC4uLvw4IyBnvd/
cquH5mcMuIlJWaRi2iIgcl8vlCj8+VtjKeuzoOZ8ej4ft27fjcDhITEzkscceY/r06cyYMSM8VBpCewrnVf/+/VmzZ
k22a6SlpfHDDz/kCLxHL/R1LG63O0/nvfnmm5xzzjlce+21jB07lhUrVmSb/5v1Q4PT4WTeu1mzZtG1a1a1f69u3
LqFGjmDlzZrYhzCfqY17eh4zqdNOmTXnvvfdo0aIFAAcOHODnn39m9OjRdOvWjSFDhpxwMawaNWqEH69f
vz7X8/jvqZPnz7Y7XYgc95uRpjv3bt3+Lms85Tzcm9ZVaxYMU+vPzPbzSfEx6OdBcnqYFrdbYFrd+Xcrsbl6ovsb+cXrPH7T5vdTWqgM7GnL6e1JAxmtEZZZqiuBEiEdakaKAhLRyFnrbuPPy8mqQLxrMoJW8ZtR9GpnBdAj3RBSsVJGKdDSFlADKNZ8mo+XRSSkXuVkHgM8I6eJMq4/UGmHOsVJ8kiwHnsKi5tg0QNiwQGQ4VpO4vJzkW+KFwHQzFiFeTAOnBsZBbGBQhmGYBYDxzqLmeyOp/mxTmOPvAj5SuSnPSQRhlVVOFsAZwqLmiMowtNiCcWf+5Cb1Rq9SJpQfZIDUWPrAWV6mMpxSvOoP0dMhDm5dVgHhtGKSZMT6WDGDzdXHfRxNbGMGeZFZqsoSKWBgzhvrGMjMzMvfxGSmUA9XrPRVcYzMiNsgS7JSA4MUs8BLnGCKX6vK3FLaqLzimIvBUp81OHFXr16Z91PNTZhTCSOpcZB0M/4a1kXYPaCKCsHGBKLu8NT9Ks4jvWkhRi+Fhag9WwbqRUZ7jbyw1dvdWyI/qmJkFPQ7Wh7JR8K7TCGSqG9iLT4Nn1BJp0m8llfBpXWi9aTdGNi2ebvGkt0XKQOrG18uvq+l9TUq3epxlv0XMzlPXtb0yD9bYx4n3/VHfZHz7Wl9k2ckdB0JxgrH4+wnB1bXkvu30nPNXHC7VSdZGqQxHmNxPDaejp9rB8mYFkgOlEYO3/S+eLjWTyuoBgDchGYnrGaqCVJWtZOpcRSi+G4OHEE/9rTz9zPE/Q3+vI/XvS+N92nIPmoe7DkhxOTNsnn7JrGtB0Tl/g5bQK5R/z/3zjLzxscPN0wsY4C5w2ppsKkupfXf1IWJcjQZwFkQxuhhhTJ44w9LjPRlpf+SzHGbTkTxwfyi+jzimhp5aUxvAzqiBzIMIs4lE5I1/dNNb9NNhMo2ZyuOCcfVvfLBhMXhJjM4ni3AiCMX0zpyWMzZ+ZUZtvORkTfTjpL8/V5LRJvpl/3HIpvvm2/6iV8z2DVN5PFWbJ6iF9LYOsZH2+SNBOn7AFIEbIvSnJvnO+4zCYDE8mG45zpETGRO3ywmUDYpQ92ZmhOHqFdLLqpvR9nFcOCZklr8WbhQHEXI5x+CTA3P/FS/0vCFsUwDFO+CdY8Ggdgh+zb7D/+eVP2JXlQt+fv31CJ9Qmgb+hoWrIq5mzPW3JnJVoF16Kv4W/8qZEzgW+cjKqMNZfQqy9ReNtw/DZahLiwl5S4gmC6tJl4zZV5P5K53TCmI3WbdCcBQWIQN6/8iq/rsyZy6MGuVIiO5ZYwrFMUH1wh2qqqDQcmAPkiAbGbbnzkLyBcXNeX4k4Gp+U4OdMHNfDGWMFBLy5JoTXpKRkeN+BkT3FZZcwKfVL6Wwqqgw3dIZfy3G71lXi7cfZMVHzL42I

+zb+/HMpkZGR7NuXgtUaycMPj+Cee+7hlltup379ptnuJ6O9Sy65nK5dL2b58qUsX/4n//vfO/zvf+/w3nsfH/f
6Ho+PtDQPy5evYcuWLTzyyLDwtdPS0lm8+E/OPvsc1q9/ieuuu47WrRPo2/dyoqMrFfj7WpDfO4vFoGLFqFy
fV1g+gWDQzFdYznhNYSrN7ene1F5xa6uw2yvN91bY7ene1F5xa6uw2yvN91bY7eneSmZ7RXVvWf/dvXtPZs
z4gQoVKtK9+8VZnoN5837j1VfHcuWV19KpUyK1a9dl5sxp4XM2bNhAbGwsf/21+pj3snz5n6xevZKrr76ejh0
707FjZ2699S6uv34gCxcuwOl05Xp90wz1we8PEBUVzYcffh6+bnLyPqPqKionE6nXz55WT++28Nv/wyl3vvvZOh
Q0fQqVOXgn4rC/17l0HDsEVERERERApB9+4X8+uvs5g16ycuuqhHtucWL17Iued2pn//y2jSpCm//TY7PCd4z
ZrVTJz4Be+9N56UlBQmTPgix7TUrVKjIxx+/z4oVy8PH9u3S2pqCg0aNDzu9TPUrl0Hp9PJzTANi9exfXXz+Q
v/9ey+TJE3nuuVF06tSJu+4aQvv2HVm/ft1pfHeKH1WWRURERERECkFcXDx16tQlOjqqacuXKZ3uuX79KZ3uuX79LefLJ4
Vx//UACgQDt2p3NnDm/kJKSwpNPjuC++4YSFxFPiBFPcuutN9C6dRvOOKNJ+PW1a9dh9OhxvPPOG+zZwe
n00FUVDSPPvoEtWvXzfX6WQOz3W5n9OhxvPrqWD7//BP8fj+33HIHLVu2plGjxvz551IuvzvvhibDYHVapU5bL
Lriy0964oGKZpFk1Nu4TYty8lX2X/uLgYkplOF2CPyk57uje1V9zaKuz2SvO9FXZ7uje1V9zaKuz2SvO9FXZ7ure
S2V5pvrfCbq803ZvFYlCpUnTuzxdIqyIiIiIiIllmMKyiIiIiIiyFEUlkVERERERESOogW+RERERERE8mn16lV89dVn
lC9fnoEDr6FWrdpF3SU5zVRZFhERERGRAvPmm//HoKULAJg5cxo33J33HAVN954NVdeeSXr1q0BIBAI8Oqr47j6
6ksZOLAfU6ZMDL9+7dq/uPPOQdx449Vcf/3A8LZGGbxeL/fddxe//vzM01l3HVVQOYO3d2j
nPWrv2L88/vyIEDB054T3Pm/MKoUSNwOp14PB7uvvvtWNm3aCMCGDf/Spk0bbrzx6vA/W7ZsCj93zz23cd
NNV3Pzzdexbt1aADweN889XH4z7h/S9fvozbbruRSy65hLvvvpXt27cBsHPnDh54YDD
XXHMZt9xyPbNm/XTM+3C73bm+L506JXDZZX04ej3j3bg2BQICHH76P/fuTT/ielUSqLIuIiIiISIFY
vXoVmzdv5K67hrBlyybefPNS7V3n//NPV3n//MypXrsyaNcsYPnwokyb9wNSpk9/ZkT/vd6yWlbWbeWmAjnpqQ
OiBEP8+ijj9OuXQf27NdZ2cNoEHX0qxZC2rVqs3ST56aXn2bx5E337PDxT9E9+lsa0cXdGNs+SaXBtbq2l3P3Ng2
2Jj68CwIEDB3jxxH4fL483df06d9zww03Y7FYqFq1Gm3atOXtfOvGxR/5XOMlmm7bxFnNnTh7ep22Ku1PnN
uwxOnbsxG+/zeapp0by+eff8PHHxAIBPj44Bj5ZcxH+2vGOgMAArFqa1gt69ezNkyMPZXuN/TjDZ+bahjsFV
O4wb9zwvvvvfQazz77JG3PzvHbnMhhxqtSLGowOnUYXqqQWQJr0BsVwNUL3j44cvmWzgPVCbf+O+76Y
psuUE4aPBnc+hYZt7ua3p/XmLVC4wZPyvNemo8xhK6FBBwN3bcBV0AvPUWTkbsE+OWDKNQkKyxOv4UKOlYZSrDxBAaNpJR7azDRHBHovCYHpBKWd72N
xvmxCn+fsqBLe3q3e0WaLelt+zAvYU83DjaW3HjhpvpQAQkbtFlRnyLEPd/fjmIBFcdpJpCpZTBncvI2e+p0+unntvaE/EYfYpqJ8vJ4+PS+CUUJ5Ktq2K3qbo2DEEF1ywcOn7hN
//17LxRf3ASAyMoqzzkpg7txfc9zH3Lmzc31fAL p168nMmdPDPX69cuZYGDRsSFRUPta69Vls2rSfr/75O8/vX
0mhsCwiIiIiIqfd4cOHWbnynyT9q5qEAatWqc845oaBomiajR4mU6u2O129uzZHa5mQigU7tmzB6fTSe/e/TFc9yTMQiQ/e/c
LHp06dRFpaKs2btwBg1Kjnwtc/fe2Dhwv/fe2/zRr1pwOTrm+d7uuede/Kla5M133e+Ktt17jhx++46677mHkyOG0adOW7du3k+DQn9IWkQxrlg4QBYsxA0Ip5dWQOoF4UhuSAthC52pz
DQ9Xdrki6NWrF++++wkjRozixRdHhs27dGrZu3UylSpUYPfopbr75724v7Ou67724CgQAA7dufTe3adQDYtWsnEyZ8
wfnnX3jc+9+yZQsul4nniUfv368cQTj2KzhQYON2vWgmnTvsM0Tfbv388v/R9zOvn17c9zH0e971vcF4KKLev
Dbb7/i9XqBUEW9f//+NZ8+O6TkND+mEPbSzqFZREREREROe22bb/8B2+8ROR2ZWdv99xiwkZsZWtG4fjbKbvHbzQJZNZ6pXN+/6+9+ke/7W7ODH3t14Z3xFuuvK
aCJYRjhc0zTxGLJHlXGj/+IDz74Oxz74H88/SGKd8FbRPitDDHzmzsfH/nz1zwhH7gdAsn+uUNwQ/fCR/AhZTIzzQHLd3uO/0//snb0gkt6cNZ
8881XADDz00DCnUeYarFYrdevWY8XSrQOcZm5QkrXk6XbNi/ezbQkf8Akw+Gbfp3rF7mqqz4YnXz+i4jgd1GrmntzNSmWsrhvY2/zvTuL8eGuMJj9G4m/etMzO9C/bmVzsmi3Tud8vl4TEdzQlOpsqsxzaaVXVG8tnfV4/hfpTvXFGjJFkNrfb3
4NxzOx/3/gMBP/PmzeWWW+5kypQPtG3bnhEjQkOO/R4x4kk2BnNnLDDVcyZsxTnHNOpxzfBwhV1Y/1vmSIj
a1E06bNmT//NzweNytW/Ennzpe1zKdaterhedmlicKyiIiIiIgUAINgMPviUKF5sXOyKF5sYOwWi188sknxMTEAFCSlSlX2
7k0Kn7d3bxLJx8fFAaAGvJ54Yzs8/z+Tttz/MMe/2aHv3JmVbXGvv3iSqVauW7fr79u0i6Ler1Ll6eH36PYYwWH/
bb7wagCFDbg8vPJabiRO/ZOfOHQC0aNGK6+JWTHPNxN4MjlcH3jZhAV/j/ZxycxX+3gwvN+bbe/aW9QDYLzn4fflS+LZqcSaNY9Z9TU2tFj5WV1ORJtQlGL
DDAjh3bAfj555ncf//d3HHHYK6/fhyNOhBAmwJiZcbhvIKQtzIaK/x/ALHhhZ+fyl06jCfBgKs8+fG6/nQx+BJ988hXPP/8yhP/8ffd
kaNWjnu4+j3PeN9yapHj17MMndOXOXcL66V66xcq1fhmjmdyIXKE3tWc5zfTDju+1dSKCyLiliMhpV6
NGTfbv34fH4wOy4E4stDU7SQmns+oUaNxsOuxx/JI+ZpcGzbaceIy0tIbff/oBq1aq
fsN3KleP46KPPw/9UlXBMHmTfhH165d5+zIvhPbxsK6rQ+LuoZ3lQdTS6mixX8lKdqnXbqnm4++kU/DD

FIYXLvyDBg0aYrVamTdvLhMmhMLirl07mTPnF847rytnn30OO3fuCK+AvXz5MsCgWrXqzJs3l1deGcvLL79Ot
249srWV2/136XIeq1atDIftOXN+oV69+jidLt5//39MnhxaUXzLls38/vtcEhPPz3EfnTolHvN9Ofqc1atXMmnSB
C6+uHeOazi/mcCu775lV3xbnur3EuaOHcQ8MLhUBGathi0iIiIiIqddTEwMLVu2YdmyJXTseC7ffDOB3bt3Mnf
ubObOnY3NZsHvD/Lqq2/Sr99lbN++nRtvvBq/4DMjDrm89OmfUfVqtU65L2MHNwvuYYDx4
8mEcfHcG1115BMBjgrrvupUaNmsd9zd69STz00L1s2bKKnvbz+ONxx2HTzxbYMQ1fRvBU5v8Xc5oaPPq9Efu
eOIZXn31Bb7+eiLBYJAhQx6kbt16vHZx48bWb/+kcK+nhQDNAdpPCLEaGbR3ZnXQnR8t6ov48j7gSEHfwXfzBlWWvdLj
CDjzB8+FAgSEREFE8//TwAd999L08//TgzZnyP1WplxIgnqVkKgBjxjxNkyZN6dfvMm6++XbGjRt93PfF6XTSq
VMX1q//m/r1G+Z4n3zjXuHrqo042G00v8c2oMO/c+ixagpRz47Cc+kVefr+FFeGefTGWZLNvn0pOYaPHE9cX
AxJSYcLsEdlpz3dm9orbm0Vdnul+d4Kuz3dm9orbm0Vdnul+d4Kuz3dW/6sWrWCTz75gBdffLVQ2svNybb17
LNPcv/9Q4mMjDrm89OmfUfVqtU466yE09LeySqq9/KA2+DGp6axe/s8fN3/D4A7fx7DsO8fxTQM9u4+eFrb
O90sFoNKlaJzf75AWhURERERkTLvzDNbUbt2HRYsmF/UXCk3t9NQkL7XIMyhLaMOjoolyUT/7Gwe9NP+Dq
NxAgGAVhVqy0AwRNU7ksChWURERERESkwgwc/wNlnn1PU3U3cg3l8tF9+4XF3U3irV9bhu+3u9SzmKn/9JPA
VhVsy3BiAhSRzxRxL07dQrLIiIiIikm/uQGjbqagIO2ce+heAg1GxrBv3YYmfrwwKyyIiIiInIS0nyhf7vKRVB7zC
Ph4yvaX1JEPTq9FJZFRERERQk39L9ocpyhA0aVwxiMUILI69LLh0xs3TchYiIiIiBSqdH/o3xE2E5cN6pYLLfK1
NtlahL06fRSWRUREREREJN+yVpYBmsaGwvK6ZCulYYNihWURERERRHJt/RA6N8RtlAybbhobOnDQa/DRGg
c3zIzkq7/tRdW9U2Yr6g6IiIiIIhIyZPuy6gsh8JykyOVZYA3VjgB8AdhYGNf4XfuNFBlWUREREREPLt6GHYjSsG
wot8AVSOCPJQW09RdO20UFgWEREERGRfEs7ssBX5JHKcoQNzq8ZOtinvo8JvVJpEx8oqu6dMg3DFhERER
ERkXzxBSBghirLriypckwnNyk+NzGOIurYaaTKsoiIiIiORLepaCccacZQDDoFQEZVBYFhEREREkXxKO7K4F0B
kKR2vrLAsIiIiIii+eL2Z4ZlI60UbKp8DArLIiIiIiki9Zh2FHKiyLiIiIiiKQ7s98HKHKsoiIiIiiAik+1
VZFhEREREEclGlWUREREREGRo2StLGvrKBEREREREGyh2VtHSUiIiIiJC5jBsAxOntWj7UlAUlkVEREREC
Rf0o5UliNtYBgnOLmEUlgWEREERGRfHEfqSyX1iHYoLAsIiIiIiSL6kHaksl9Zto0BhWUR
ERERERPLJfaSy7FJlWURERERERCQkxi2iSsiIiIiiiglZhi4iIiIiiOSgYdgiIiIiiR8moLGsYtoiIiIiMgRGXOWI0pxZ
bkU35qIiIiIiicLr/vsLLmbxhYD9yBjLBceivLCssiIiiiiJyXKYJI3+P4LAPPOmO8PHSHJY1DFtERERERESOK8UHh32
havKyPZk119I8DFthWURERERI7rgMcIP16bnBkjVVkWERERERGRMitrWPYFMx+rsiIiIiiJlVtawnJW2jhI
RERERZEy62AuYVmVZhEREREREWERERSmzcqssu1RZFhEREREeSmzcqssu1RZFhEREREeSmzc
g3LdlWWWRUREREEpIzKWODLelRmjrAWQWcKicKyiIiIiIiiIHFdGZfnM+MxjDouJtRQnylJ8ayIiIiIiInI6ZITlZnEQf
WTodWkegg0KyyIiIiIiInIcQRMOekNhOTYCGlQIAKV7CDZAoa5ddt1115GcnIzNFmr2qaeeIjU1ldGjR+PxeOjZ
syf3338/AGvXmXEiBGkpqaSkJDAqFGjsNls7Nixg2FGjsNls7Nixg6FDh7Jv3z7q1avH2Lfji2avH2LFjiYqK4tChQzz00ENs3bqV2NhYXnnl
FeLi4vB6vbrkYO3YsDRo0KMzbFhEREREKbEOeyFoZoblc6oFWJFko2lsoIh7VrAKrbJsmiabNm1
1i6tSp4X8aN27M8OHDefPPNpmEyZMAGDgxgTxFcffXVzpgYrRqV69mzpw5AAwdOpTHxUrFcffXVzJgxgx
YtWvDmm28C8Morr5CcXSaJ4Dik+MkMD06dO5/PLLefbZEEP348ERERTJ8+neHDh/
U3NPPyUbdUnjnXXYS9KniFFpb/++8/AAYNGsQIzdrzcpCKB7CfbD7I8F+Z6vK6lTpw61atXCZrmwXIL
Vi3LSIiIiIiUqJl3TaqYgTgTYLNCichBnKR+GXWhh+dChQ3Ts2JE33+4iC+//Jze9Ppo7du3ZtKkfTIDIuwBCyiMO+gKxEIYXU1INgSHY3tJ1
bMfj4uLYvXs3+/fvJzo6mO4kOO01NpuN8OxOkhkdzAowzAkNm7mtXbt2FcYTi4ti3mHhHHoiiIiIllHhZw3Ksqksqwg7UsgKbc5ymzZtWLBgAQDLly+nYcOGmc8lJSUxePBg/H4/Zm1tRERERESEiIiIimkCwXsDIKhJzQNZfGHQd95hsybuppddMTF7h0zcO6M5ykKjP4gcKe6ur2x3Q6sNUnbdifIgZTauVd
bK8031thtLd1rx0IlZRjIyg192axGMctjhbaMOzDhw/zwgsv+4PF4SElJYkLzTzwwANs3LiRiZs
EwgE+P777+nSpQs1atTAbDawNhbdAwSQ0OAKZzT6hrYXOWyNMRzAPpQUQQ+3MKCYQAKPBDoBzT0tbdRN4MMr0vUKEBqM0XAXqYuxBKjEdeXdmZp7ZYAQEeFYHjyNiDgjX1ex0BQkKOhOmkvcD7cL00Ptgjs
NhIQE7HY7iYmJTJ06Oq1QvrtkVERERERERDrEGwM1Yt8xK6Z+iYSD9iedThYKm65wq1izBLp8frm1jl3fYQDfE1Vpl+fzzz2fFihX069eP6Y0ZYKeQrHsmJ0Vj3K4TJENiUrzVJiT2n7pfFbGh2wR0m8+lYPiFdaFNdZ02xlzYWxiOJvvWP6zC+L29IZoY6WWWOe9YZBoUl16X+I+6XETlgT7MQCQTMKwyMmQCNR7UtKBSoLpmy/+i+Xvoc4WVfsZSKLH7LtE1XLjOpF4IZCx9dlltWXpdc4h6uaNnjDpCNb6nj3wF7zvKHDiWF6x4I8S8ZiK7OuyK1cN7TfyXBqZgzc7+cKHBRfGBhqOizQZZ8vqUJvK0CBEHJ6qsJ+KfKEFbyqGRmVFr4JIXGTczG7tYy8sq0wIipOm7p6ZlfuSQxYAbd9LRcD+XXHDJMa0etPwb6jmfjJoh1NfTzPJ2umBD33+7xuybbPOmp3G4MWvpZhkvDm9Ayq1htNdR41IMyxy3a5mXGNtNH5hOx6K4CXkTkZwUlLxxLsh5wawfoivWdpTw9ALBuJTIFKAndpYGtsHZdbgV7tWdAXVIBCTkiiFhHGNsrJq8mrdqmqdlIhJMgNzqMPtV3Yp2TG7qCZBF3jRvSNiOPoqQSVDl9nrA1XLmwmmWy/wAbI9vxFfeEzFoU0Z3zCnvjmxqLZGBWWjdmbFsyONHYMKc+i8t91Fz/NR1hGSmjyF+j3XDDEAa8Oji7n6LNbhE6lPjGzGzKobXR7DOAZLvq7rUOkkmgjLHvkN3J7JOo0jj3exHq1wNXjKrn4q0sQxR0V1vRBnuxT0EoNFgjZL0UEyd3mxF0r1gsrOr+bk1cypFypiyvDHBXZ07Q3pn7r9u80MgVZZCwP61y4Ya2XC3HmnS0ZQFaDwTEVcJJTexb8yxemPlRFq7yHZXLjgpy0dW31qYDtn2phLYgMoIZtZvRcdjzkVo+o0ihlPNNmCHw9Pu/VRjR2mmemBG2HzLlR5WmL+AnhcB3NRXNOtcbUJRJ7XXOq8OpWSQAWmoN2NE0ttzcmbHEnhdQIMaaF4OSY6h8ZAQ4yqVpx4pcnQN3HT8oRswL1VEn1kqKG9HQ4tkrF2dGFBgcYvUYH7xmN9YobAk5eHWrnLWeWFUoPZgmJAX3yIZLjxZ4M6rNj2rqkFDnBPTUxqQd80ynGdMvJ9dxDeZzdEXXKKgdjSPbDUOUEd1xeQ0QBDr6SeUDDrXDtlxUT2TJodAYPLMvCAlp4kuUKAOhYUpBBWuJzhtVL8wR0SjnJ8sWzxpT7dbaVmTc0dc3Ua6yAh5AXTeM0lZhfXU7WTKZ5uRw0lyA3zTXUGEHNiihqIBIq3YBXRsrvbiLgsrhEEemRqeXh9vrBgvlbnE=

fTZs2bRgzZgyDBw/G4/GQmJhIjx49ABg7diwjR44kJSWF5s2bc/311wPwxBNPMGzYMN566y2qVavGSy+9B
MC9997LsGHD6NWrFzExMYwdOxYIbVf1+OOP06tXLxwOBy+88EJh3bKIiIiIiEiJcsgLKV6D6tGZo2sz5ixXcOac
FluaFeo+y/fddx/33XdftmMdO3bk22+/zXFukyZNmDhxYo7jNWrUYPz48TmOV6hQgbfffjvHcafTyfPPP3/ynR
YREREREESnFTBM+XWdnxiY76/dbMDEY2d5Nv4Y+IHtYLksKbRi2iIiIiIiIFD+r91l49U8Xf++3YhIKxm+scJAWyso
KyyIiIiIiIlL2zNueOeD4otqhhLzfY2HCegcAB4+E5fIOhWUREREREREpI+bvDIXlxhUDPHuumwblAwCMX+sgxa
fKsoiIiIiIiJQx+9IN1iZbATinuh+LAbe39AKhVbA/W+vgkFdhWUREREREREMqQP3Zaw4/PrRaqKJ9f00/jiqHHH6
1xEDQVlkVEREREREKQMyRiCHW03aVE5FJANA+4/ywOAL5i5VZTCsoiIiIiIiJR6gSAsPBKWO1T1Y8uSDhOqBBh
4hjfb+eUVlkVEREREREKS0W5Ns4eCR+cjnVPfneH5waw+1Y4Lhr1VZFhEREREREkVJtR4rBC0tc4a/POTJfOSuX
DZ7qmE45h0mNqCANKgRznFOa2U58ioiIiIiIiIJQWC3dZeXReRHiV6/Nr+YiLPHbVuEXlIN/2TcFlJdsw7bJAYVlE
REREREKSMSPHB8CxB+arGXoa09hz3NdH2wuhZ8aOwLCIiIiIiUkZMWO8Iz1Me1s7NZY18Rdyj4quMFdJFRER
ERETKphQffLbWAUCjCgEGNFRQPh6FZREREREREkTIga1X51jO9WIwTvKCMU1gWEREREREp5dz+7FXl82rm3
CpKslNYFhEREREREKeU2HcrcU/maJqoq54XCsoiIiIiISCmX7M5MxzVjjr1NlGSnsCwiIiIiIllLKZQ3Lsc5gEfak5FBY
FhEREREREKeX2e7KEZZcqy3mhsCwiIiIiIllLKJbtD0c9uMYmyF3FnSgiFZREREREREkVJu/5Fh2LEuE0OLe+WJwrKI
iIiIiIiEgplzFnuaJTQ7DzSmFZREREREESklMuYs6z5ynmnsCwiIiIiIllLKhSvLCst5prAsIiIiIiIiJSiplm9jnLkjcKyyIiIiIiIIqVYq
h+8wYw5y9pjOa8UlkVEREREREEqxjKoyqLKcH7ai7oCIiIiIiIjk365Ug0n/2mlVE84qDxG5pLvkLGFZc5bzTmFZRE
REREESkhNlwwMI9v0aQlG6BvyDCFk2/Bj4eOMuTYx/l/e7MAcWqLOedhmGLiIiIiIiIiUAL4A/HvAwvSNNm77O
TIUlI9I9xt88beDv/fnjHjJniyVZe2znGeqLIuIiIiIiIBQTpgkBE2xHZd4lu608Nt+VLSAD3NXKQ41YJyN+DX29K9VC
k9jsi3gla87ySVFlWUREREREpBj4Z7+FS7+PYsB3Ufx3MBTVTBM+W2fn7l8isgVlp9VkRHs3g5p7uah+5jWS0o
2jLxte4CvabuKwFuw9lCaqLIuIiIiIiiBSxZXusPDAnghRfKNgOm+fi4+5pvL7cyVfrHQA4LCZ3tfLQJj5A3XJBouyh1
1aKBAMTE4O9xwjLydpj+aQoLIuIiIiIiBShBTtDQTljL2SA/w5aufKHKLanhqrJVSKDjO2STtPYnPsk2yyhILzPbRy
7snxkznJFl/ZYzg8NwxYREREESki/x6w8MhvoaBsNUyeODudtvF+gHBQrhUd5MNuaccMyhniIkJV473px1jg
K6OyrMW98kVhWUREREEpAjsTTe4b3YEqf5QmH3mHDd96vt55hw3FZ2hYFw7Jsj/LkwjPvL4QTcjLB+rsp
wRlrXHcv5oGLaIiIiIiEghSvHBl387+Gytg8NH5ijf08rDRXVCFeW4SJOPu6cxd7uN7nX8eQq5lSND4frosOwPwk
GP5iyfDIVlEREREREGRAuYPws9bbPy0xcaCnTY8gcxQ27+hlxuaebOdXz3a5MrGvjxfP6OyfMBjwRcA+5FVrw9
6DEyOVJY1DDtfFJZFREREREQK0PwdVl7508l/B7Pv23RGhQC3nekKsAYfI+fo6XzJCMsA+9wGVaNCX2cs7gWq
LOeXwrKIiIiIiEgB+XiNg9eW8O8Nfl3eYdKnpp2stH+dUD2A5xZCcoXJE5uJfSemZYTljvjJoznJ+KSyLiIiIiIiIichE2HD
FbvtXJeLT/R9pzP+wLwyZrQHslRNpObWni5qrEXpzXnuacqa2U5Kd0ChMJz1rCsynL+KCyLiIiIiIiIjkw+q9Ft5d7e
T3HaE41T/Jy4gOnhznLdxl5aA3FFYfaefm4nr+AutT5axhOS0zIO/PGpY1ZzlfFJZFREREREETyaONBC7f9HIk3m
BlC5+2wYZqeHPOOf9wcKjc7rSaJNQsuKENo8S6rYRIwDfZmCcgZYd3ApJzCcr5on2UREREEREZE8enulA2/Qw
MCkWWwACA173nI4e1J2+2HOtlBtslN1P1HHGKZ9OlktUOnIMOu96ZkxL/3IHs4uG6dtfnRZobAsIiIiIiIiIiJylN+2
W/linZ1DWXZ0WptsYdbWUOq9uJ6fxzq4w88t2R0KxtM32vi/RfDpOgepR4JqtzoFW1XOkDEUO+swbPeRpl
1WVZXzS8OwRUREREESkTJu1xcZbKx10rhHg9jM9fPm3g9dXhFaw/t8qJ9c29ZJQJcC7q0KLddksJred6aFalEl5
Z5CDHgtLdlupFRPksT8ijlw19Poom8m51QsrLAcBK0npmWE5o7IcoeSXb3rLREREESkzEpKMxi1wEWa32D
TISvf/2djvydzAG6Kz+Dtlc5sr+nfwEeN6FCl9qz4AL9utbBsj5VD3pzjnC+o7cdVSKkrY0XsrMOw3aGR4rhsqiznl
8KyiIiIiIiUWa/86STNn2X16CNBuXJEkDvO9PLVejv/HMjc6ynSZjKoeebY7IT4AL9utbPPbWHfrtBrr2gGLcqnsyf
dwqUNs4zjLmAZYfmg18AbAIcV3Blzlgtgu6rSTmFZREREREETKpCW7rcw8smL1BbV8tIoL8PZKJ1Uig7ySmE7N
GJNLGvj476CFXakG+9wGLSoFiYvMrNImVAlku6bVMBnSwcDlKZyh11lVztKvvekG1aNNVZZPgcKyiIiIiIiUOVs
PGzy1wAVAhM3kwbYeqkSaXNHIh8UIrS4NoRWkG1YI0rDCsa9Tv3yQis5guCJ9YW0/tcrZSUoqhJs4SlxEMPw
4KSMsq7J80hSWRURERESk1EnzQZrfCK8QvXiXlf9b7qS8w+Tsan4+WuPgwJGAe9uZoaAMYM9nqDQMaFsl
wM9bQte6rqkXKOB9onKRMQwbQttZQZD0IwXuCFWW801hWUREREESpVFu6wM/93FAY+Fs6v6aRQPn
66MwCRUZV2wKzMGDWru4domvlNq77qmXlbttXJBLT9NYoMnfkEBiXVlBuIDntC9ugOZ+yxL/ugtExERERG
RUsE04Yu/7bz6p5OAmRmMF+wCMHBZTSJsJvs9FqyGyaPtPPRreGpBGaB5pSA/9Es95eucqsgs1eNUX+j+V

Vk+eQrLIiIiIiJSKnz7n42XloXmITutJhfU8jN3u41Un8EZFQM8d46bmjFBViRZqegyqV++6KrABSHCBgYmJgZpR
z4D0Jzlk6ewLCIiIiIiJV5SmsHLR4JyJVeQV85Lp2lskDQf7DNiqGZJw3Zk0a62R61gXVoYBkTaINUfmq9tmpnDs
FVZzj/LiU8REREREREpvkwTnl/iJOXI0OMR7d00PTJ3ONIOZ1UjHJRLu0h7KBSn+sCT5TMBp8qk+aa3TERERE
RESqSdqQazt9pYk2xl9rbQCtTd6vjoUrN0Vo7zItJuQjqk+43wEGyACKsqy/mlsCwiIiIiIiWOPwi3/BTJ7rTMknF
5h8lDbT1F2KuiF3Uk4aX6DNKzfGbg0jDsfFNYFhERERGREmfVXms4KEfbTc6oGGBwa0+27ZPKooxh2Gl+sle
WlfzyTW+ZiIiIiIiUOPN2hJZ3NjCZ3CeVimU8JGfI2D4qzWeEt40CcGkYdr6VkWnuIiIiIiJSmszbHqr7Na8UVFDO
IjJjGLbfCK+EDeBSmTTfFJZFREREREKRE2ZlqsOFgqLLcqbr/BGeXLeFh2D5wZ3lrtHVU/uU7LB8+fJjU1NSC6IuIiIi
IiMgJZVSVATrVUFjOKiq0KDhpfoP0LHOWXdYi6lAJdsJifEpKCl9//TW//PILK1aswOfzAeByuWjVqhUXXngh/fr
1Izo6usA7KyIiIiIiMm9HKMZUjgjSuGKwiHtTvGTMWfYEDFJ9mcdVWc6/XMNyMBjknXfe4d1336V69eqcd95
5XH755cTGxhIIBNi/fz9//fUXX3/9Na+++iqDBg3i1ltvxWbTYHgREREREREREETk5Kb7Q9keGkfM5TwA2HrSwZHeoT
Hpudf8xzyvLMoZhAyS7MwcSa85y/uX6ll1xxRU0a9aMr7/+mvr16x/zH79+gGwdu1aPvnkE6644aomTZSqpUI
B0VEREREREREZGSzzRh7nYrjmSoZTeoEW1iGBA04a0VDj5Z66BZbJCxXdIp7zT54C8HP2+xsS/dwkFv9mTcqXogl1
bKrqgsCW+fO+sCX6os51euYXnMmDE0bNgwTxdp2rQpo0eP5p9//jltHRMREREREREREEkdLnk7UOXlvuPPJVNLGuI
F1r+dmVauG3I8OrV+2zMujHSMo7TdYkH3uybe2YIB2rab7y0bJXljVn+VTkGpbzGpSzatSo0Sl1RkREREREREREESq9
Fu6y8scKR7Viy28LX/2Qei7GbHPYZbE+1sP3IusL1ywc4Kz5ArMukdkyQuuWCNCgfxK4AmENklgpyRmXZbjG
xaR+kfMvTW+b1ennzzTfZvHkzAE899RRt2rThxhtvZN++fQXaREREREREREREREREREKfl2pho8+ruLoGngtJq8dTEMbeum
YzU/FiMU8M6p5ue7vinc2sITft3Vjb182iONYe083Hamlx51/TSJVVDOTaQ983FGZVlV5ZOTp2neL7zwAtOnT
+f8889n9uzZTJgwfvuu49ff/ff/2V0aNHM3bs2ILup4i4iIiIiIiIiIlGAvLXVy0BOq1Y1o7+biRhEkVfAxsLGPvekG21IstKw
cwGLA7S29dKrhx2GBRlrtOl+OVVnWStgnJ09hecaMGbz00ks0bdqUTz/9lA4dOnDLLbdw7rnncuONNxZwF0
VEREREREREpCTbnWYw58jeyD3r+ri4xXa5xpUjTCPHZF+sq3klheSTEZVlznLGhxNaCfvk5GkY9uuHDh6lTpw4Av//+ePrF4TpwA4Av//+
O506dQlgOjoar9dbcL0TEREREREREZEZESb+oGO0EzVOW8qrHyQ0GKPEYwdllVWT4ZefqMoX79+vz222/222/Ex8eza9c
uunTpAsDEiRNPaiEwEREREREREREpG/zBFGaFIxQDNVjAtU1spyBlWWT06e3rZ7772XwYMH4/f/f7f7f/7f7ufjii2nQoAFj
xozhs88+3u3z/FPXv3xxx9MnzaamrjveZNm0aq1evZs6cOQAMHTqUqx9/nJkzZ2Ka2UOC3rNoDeGs9BwApp2d0OnoD6gFxhWqD6jFk
Xngh7du3z/FPXv3xxx9MnjwZALfbzfDhw3nzzTeZNm0aq1evZs6cOQAMHTqUqx9/nJkzZ2KaJhMmTABg1Kh
RXH311cyYMWLVqEFxd75ZVXSEhIYPr06Vx++eU8++++yzAIwfP56IiAimT5/O8OHDefTRR/P+zoiIlIiIiSSJjbDy8
tcwJQ3mHywUWpDGruoX9DLzc39/B4h3TG90jLFpSlaBy9IrYqyycnT2G5atWqpKWlnVDfHzQpKWnlnVJDBw4c4c4OWXX+aO
O+4AYOXKldSpatWpahotpSmFbkI2hmrqk+lXFdk7o9cXAvIcBcCmmgCN0tV+sJU4rTfcP3/Yp8u5S7adCqBUa4X5ELxu2bT7vOAk83rgjIV0FuX6QnxmQNqkspzAgeShHeMPaNQoFXxpNvr4yGv7Ouu5+XKr2g7/bXM2P3oZJ3tKRRMSNwHvhT1d+d4kJ4BJJ8x/ZWBxR4U61nO8knh0FxjuMSJNNdaRJSPiFl+E0BJHkSFIH+OAuddVSj1EUEptNRWq2w/fXhSeT8Aw+sBuCfc3SfB1yRuWfhuNCzzXl5IBKgfjUZRgaIhdiuNF6Em+NGo8yu3MHFqKQ1a4H0TXA3G6pTo5mppkVKrXJmkiwXA3zsTsZnV3dOa1n7iJb6ZdBXNvq9mCCkPzNn8dNQq4CnKN9WHyOEk0MOtBuGVSuNcimE4aAnimUxSfIKkRufcAVsSbtLdJdbM4D/yl3TFjGAFzVgDuwNWd0XZXIUn/1FvJbVEVIkXT6qdhzsd9ZGWSiJwJ7LcWPUUz78Ypm5VGh9PS7jd/PkfUUY0Wkvo8L7+tdhJk3HIPhHYxJ1QTIJafq9fsuGXyymn1N+xOZ5Wbh/BILD2AHMFjlgLaQjW+VAbHTtgbtYcAuPULPWJUoVyQTAVhqnSkUPFkjcpT5AfX3Ff6Z+rXrfs+uV9y9NY0QBQuwZl8Wdva9dRQHvMsWHhP9LdYgyM72DFOfrYnGhBLAf4sP5/Kn7NhzalC4uTzaKR8knEoy8oYQ8ND4yVI9p72b0Fo7gLR1hCDaWrxEpAgyAmH/7n7JCHPRPh2ZtXPtTVJ7mbhD8+d2H71m5+b/zZGAhl2oX4XB+pC4O5r1bfAFrOQWAZNc51zi1gJaPtoEaBPSzvlz2kHb82V2oKhAWcGbpxJbYBRhCfmPbS7nlqV2QAk5/yMm/DsNdSwDi34wEJV4g9CMwljBnq1o+XO3OO1M5SCtjTAgMMNLq7uYb5OaglNBlzHiUOnn9S8spKUtAP0DyQo2Y7Gnx6PVpQ6FozW8bGC6i1Xk3dqt6xbmgwDqqM3YQNqdPgfJKvnQoS/Bp+rIv9Kxd+MqP8Z/o0O6C6kjTzbTk9RNOVUCzrgcs7MG+47gbp7vS3oDjmuHtORN7S3RR6Z0HDBGqTTO8PvyFOiaG3F7/J7ehEWddQdSjQVsfr+8m4QHdgjixprO1iqGX6fk3mYb2bVb9l3/7GOqWc1dmfYkGKpHs4QJR0txVw5Yi6Lio59EoWTr4kOD5VnB0PzvsfyNJlQC+6UtrG49uZ6E8m5QM2jWb4jmnHBHN2Q/lrrnVXfUb6PMFiGTkqQ1W4tJm7EvP+1bphS16dNkfDnKzk8CIKpX1pcQJfnIqHfEcvLGQWK8hJYyNGQRfRmZMGIPQrWkhk9eHqaoj3sF6bw2c0uSpZG9M43cG8Pgv7vvYdcIRZVqOVvDIlDEH88

q+L3+0lNTaVChQpUqVKFPXv2ULt27WzXyo99+1IIBvP+SUxcXAxJSYfz1capKM3t6d7UXnFrq7DbK833Vtjt6d
7UXnFrq7DbK833VtjtFdd78wRC1d+jF9/6bQt0+9Tk3jYeBjT0YTEgxQuT/rUze5udVXstVI4webFzOpbIKB6bb
Rl0Df7YBn9sy95G11o+RnZwExOELH+uF/i9nQ7F9ftWUO0ZfifgCH/tT08nKekkhxqcoK2CVpDtWSzGcYujeQr
LGYt4paSk8N9//2G326lVqxbR0Xmrun744Yfhx5MmTWLRokWMGjWKbt26sXnzZmrWrMn333/PpZdeSo0
aNXA6nSxdupS2bdsydepUunTpgt1uJyEhgWnTptGnTx+mTJlCly5dAEhMTGTKlCncccccdTJs2jYSEBOx2O4mJi
UydOpWEhASWLFmC0+mkevXqeeqziIiIiEhxtyvV4I+dNr7faGNFUuhP+4Qqfs6v6v6Wfdfivf/Wcn3W8wZrGLW
Vts9Kjr462VTvamZ462TEo3uH1WJA4rBE0Dp9XEakCaPzRCs3HFADc199K1Ip9cBnZKMRN11LBrl+Ysn5Q8v
W2BQIDRo0fz5ZdfEggEME0Th8PBFVdcwfDhw/M9tBnA6XQyZswYBg8ejMfjITExkR49egAwduxYRo4cSUpK
Cs2bN+f6668H4IknnmDYsGG89dZbVKtWjZdeegmAe++9l2HDhtGrVy9iYmIYO3YsANddddx2PP/44vXr1wuF
w8MILL+S7nyIiIiIixdH0jTaeXOAiYGYm2LPi/bySmH4kHPnoUdfH0wtc7EqzsHi3jcW7M//8rx0TSMkAWZssu
EJGHgCoeNPnu2mfVU/P22xUysmSPsqAYXkEuboBb5cVs1ZPhl5CstvvfUW3333HSNGjKBdu3YEAgGWLFnC
a6+9RuXKlcMrXOfFgAEDGDDBgAAAdO3bk22+/zXFOkyZNmDhxYo7jNWrUYPz48TmOV6hQgbfffffjvHcafTyfP
PP5/nvomIiIiIlATrki08vTAzKJd3Brmgpp/7zvJkqyJ2qBrgy16pvLrMyeQNoWG55R0m953lpne9UKW4Z10rj8
yLINVncHNzDxfVCQ3XvayRFvEqqSLt2b/WKuUnJ09v2zfffMOTTz5Jz549w8caN25MaNy25MbGws48aNy1dYFhERER
Epi1J84PEbVIrIXuU74DaYsiGUbq5p4sVuPdarYXeawd/JFiwGPL/EhTdoYDVMnu/spksNP5Zcqr/RdhjRwUOP
un5W77PSp76PWFdmH86uFmBqn1TSHdFUs3hPy71K0co5DFuV5ZORp7C8f/9+mjVrluN4s2bN8rx1lIiIiIhI
WfXvAQt3zorggMegex0/Nzb3sjfd4LftNqGMuzlSDR9t7crz+gAeunhbFQW/2RDy4tYfzauZt4aa2VQ
K0rRI45nMVXCaN4k7Pwl1S9I4ehh2Rywcwcnx5CssNGjRg1qxZDBo0KNvxn376ibp6ibp16xZEv0RERERESoVtKQ
b3/BrBfk9onZ8Zm+3M2Gw/5rnf/OugUcVgjiHQP2y05wjK3ev4uKaJhkpLTkfvs6zK8snJU1i+6667GDDJkCGvXr
qVNmzYALF26lBkzZmhOsIiViIiIjIUf7eb+GPHTb2HHKeZ6w+3bJygJV7s5f52lfxM6CRjxeXONnntvDiEid1YoK0qx
qqApsmTPk3FK5rRQcZ1TEdqwWaVAxq4S05phwLfCksn5Q8heWuXbvy8ssv8+677/LTTz/hdDpp2LAh//vf/+
jUqVNB91FERERERERERERERERERERERERERERERERERESoVtKQ
ERERERERERERERERERERERERERERERERERERERER
ERERERERERERERERERERERERERERERERERERERERERE
Ep/2GEBVvx8Kt2bp9dk/jYMNtKFRRdjBFSaZAhjVIKBRERERERERERE

kRz2pBk8/ocLCO0le9uZHmrGmLzdNZ0Xlzj550BobnKEzeTRdm4urucvyu6KSAmUa1hu1aoVrVq1YtiwYfz55
59Mnz6dBx54AMMw6NGjBxdffDEtW7YszL6KiIiIiLDlkMHdv0ayMzU08vGGVlAzJlQxPis+wKc90vh+o42/9l
m5urGPukcW6RIRyY88LfDVpk0b2rRpw/Dhw1myZAkzZsxg8ODB2Gw2evbsyUMPPPVTQ/RQRERGRMi4QhI
n/2HlrpZMUX2gVrssaeRnRyUHyvszzrBbo28BP3waqJovIycv3atgJCQlEREQQERHB559/zkcffaSwLCIiIiIFZtth
g1+22vhho50NBzM3Ub65uYc7WnqxWo6/V7KIyMnIc1heuXIlM2bM4Mcff2TXrl20a9eOoUOH0q1bt4Lsn4i
IiIiUQX/ssPLlegfrki3sc2dfaLZGVJChCW461QgUUe9EpCw4blheunQpM2fO5McffyyQpKYn27dtz6623ctX
bG1tYfRQRERGRMmTuNisP/RZB0My+4XHtmCC96vm4pokXV77HR4qI5ii0i0ifz49WwYApzq6j7sy9a5Ug/vnROIJGFg
Nk9HnptOlpoZai0jRyDUsjx49ujD7ISIiIiKlhMcPt/4cyrdkzMW41iRbuWZFc19nJhbT+NK+asEo9b5uSQN3R
weHu3grKIFKlcw3L//v0Lsx8iIiIiUkq8s4xwUK5bLkCruADfbrDjCRh8tMbJR2ucNI0N6Op56cS6Qvsj/7HDyq9b
7QD0rOvTk8iUuTyMHNERERERERCRvdqQYvL449LhB+QBfXpzGYx08/O/CdJrFZlaK1yZbeXaRE9MEbwBeXOo
CIMpmcm8bT1F0XUQkG60jKCIiIiKnzbhlTtxHisLD2nnCi3qdFR/gkx5XXK5IkObdgwm3mnn
DoxNtaeqgcYRZR70VEMqmyLCIiIiKnxff/hQIwQK96PtrE55xkCzw
wuuwiMhxKCyLiIiIyClbuMvK0wtDQ6kruGBI69yUldwmTzZZ3t2DnV/Lx2fnqetpcSEkMeR6GvXPnTlau
XInP58M0sw6ewDXxdFK2V5Rk0YtBoXYbE/Xxdx0ukRvEZHWHMs7h3I6fRvsGxHXtzMYF4vm4pokXV77HR4qI5E+uP2Y6d+7M/v376ChChCW96vm4pokXV77HR4qI5E+uP2Y6d+7M
/JxWSOf9lAWkWIlT2F5woQJjBo1ikAg51ikAg51AawzAUlkVERETKkFQffLXewadrHe6uimfY0IkpJOfJ0bm3
m5sZm3gHopInJq8hSWP/zwQwYMhGMDDDz9MTExMQfdJRERIqpVXstPDg3gmR39vHSFZxB7mnlpVsdbf
kkIqVDnsLyjh07GDokIKyiIiISBk2f4eVh3+LwB90dERokIKyiIiISBk2f4eVh3+LwB0IVZMbVggwqLmXs+IDVHKZGkYtIqVKnsLymWeeyfr166X1
5B90dEREEiqE520JBOWAaWAyToW09XNrIh0UBWURKqTyF5f79+zNq1Cj++usvusv6tSpg8PhyPa85iyLiIllF5r
ky0M/z0ivIDXs+e6uaCWhluLSOmWp7A8YsQIAN2cz2mBLxEREZHSa1eqwf1zqn07lGzkVfR
URKmzyF5XXr1hV0P0RERKQM25du4LSaRDtOfK4UnL3pBt/8YyfVZ1DBZfJ3soV5O2x4jsxRfuAsj4KyiJQZed5
nWURERKQgLNhp5YEZE5EZR3mnzeM42KruPvzyunT9AM7ZUc7YC/91t4YE4Eu9Msxzx34BlermzsK+QeiogUn
VzDcosWLLyLZg7y6iKl2+40g5HzXXiDBknpBh/85eDBtp6i7laZMHurjdGLnexzW
6gRFSTZY5DuD/2957CYeIMGUTaTLjX9dK/j49zqqiiLSNmSa1h++umniY6ODj8+XlgWEREPytyS9/EIb/7uKAJ7O
SOfEfO1c19lI9+sTV5XQ/TN1g558DFnakWKgaZTKouYdaMapMH4/bD2MWu/h+oz18bHtq6HtgYHLfWR6u
buzDHQC7BWzHLjSLiJR6uYbl/v37x8PGDCgUDojIiIZYPbD08tdLEiKfSnSPuqfhbtsuELGoxe7MJmgSW7rdx2
FlzbgBz79/oCcO/sCJbtyf6nzIxNNq5p4uXWM704rYV1NyWHacIzC13M2BwKyjF2k/4NvWw8ZGVvusHtZ3ro
dGROcoQm64lIGZfrZ4XXXXddvoZXL1++nGuGUUpWusZmRdUmyjrSPLOHpEdnDyJx2DNDYdCDzCRjXmtGqheb0D/
rHTxm/bbaT7DV5dBP9bFVr1y+2HA55Q4Ht2kSsclMs7TM6oGAp4vqDBR2ucPDQ3Ao9GDecwdYM9HJRbV
g7wVa9UhrTx8nJiOuN7pIWDsoiIHKeyfO+99/LQw8GbNmtnO2bBhAwsPO5MkcP
HiQZ599tsA7LCIiIsXXrtRQCbhqVM6h0BsOWPh4Z8X7GdPJjb6Dmn1STCZnLA
Y+G91U5+2Wpj8yELAdPAZTVxH1mduUWlAG93TcNlg7/2WRiz2MXaZCt/7LTx0NwIxnZJV4X5iLXJFl5c6gSgc
kSQsV3SidViaiIiuco1LCckJPDtt9/yzTff8PHxPHHHDGD8rVoqfJCJYf2CexeHWSYYZDof2
exARESmr/tlv4cYrFIzFNeOW8dNpXDVUpPQF4Z5WD8WsdBM3M8dRXN/YypI0nPCe2bjmT/3VNZ8NBCxfU8
nPYC3f8Gs2eVPjvYGbizQjK1aKCjOuSjuvIXzPNKwV558I07psdwdI9Nv7YaePpBS6eOdddOG9AMTZ7E9z5c2
R4r+RnznErKIuInMBxZ6M4HA6uvvpqrr76anbu3MmsWbOYO3cuEydOpFq1avTo0YOhQ4cWcHdFRESKnz0H
zh/XiH/hbB+xelsTvN4JVlTjYeCoVdi2HStZafG5p5aRIbzHGd1vEBWseHQnZFF3w+AO6bFvq6TXyyAWFeQnak
W0v0GNzXzUCkie+CLsIWCesZ85hmb7dzY3EvDCjnbKism/2tnzGIImAYGJg+e5SGhioZbi4icSJ6X0y6Xhu8716
UryL6IiIhICRI0YV6nbbbSzZnflnRarP4JrpkeHh1gDNYgM8frY7X8G1USx81D0tX/2KsMFjHdxc+n0UQd
Pgw78cPFFtGq8sT1tt5YYkLAKfV5KmObrrW9hdxr0RESgatcygiLiL55vbDbz8w4I+7Uc2vLhX8
5wUI6ymdzY3Mt1Tb2Ftg1RrRiTi2r7mbnZzk9bbNzR0ihzW0plDcrlnfBKYhpnVi67FXYRkfxSWBYREZE
Qdh

2LxQUAbYdChzPvFtZ3oYeIYPuwV+3mKjW20/VzT2Uq4IljW5qbmXmZvtBE2Dj9c4GNnBU/idKALpfnhpqZPJ
G0JvejmHyWcDDKoaCsoiIvmhsCwiIiLHZZqwz22wI8Vg/Wb4dm0ka5JDAblZbAB3ILQAV/NKAS5r5MMw4LY
zvdx2prdI+92wQpAuNXzM3W7n+4127mrlLfWLWm09bHDv7Ei2HA6V8Ms7TN64II0z46NISirizomIlDAKyyIi
IpKrn7fYGL3YyUFP1vHToaDcvFKANy9II9IGu9MMKrrMQhtmnVdXNg6FZX/QYGWSlfNqld75ur4APDIvIhyU
z4r381RH9zG38RIRkRPL06800zSZOnUqu3btAuD999+nd+/ejBgxgrS0/C26ISIiIkUvxQtfr7fz3CInGw4c+8+Bf
/ZbeOIP11FBGWrHBLmuqZfXzk8jyg6GEdpXuTjuZ9y8UgCDUFhcm1zMkvxp9r9VDtbvD30TLm/k5a0L0hW
URUROQZ4qy6+//joffPABH374Idu2bWPcuHEMHDiQhQsX8uKLL/LEE08UdD9FRETkFJkmrN5nYdK/Dn7ab
AvvV/zDRjvD27vpVS+z6prqC1UpM/blvbOll/rlA7SvH0mEN7WobiHfouxQu1yQzYesrE0uhmn+NFm2x8rHa
0JzlBtXDPDAWR6spfuzARGRApensDx58mRefPFFWrduzdNPP03r1q154okn+PPPPPxk8eLDCsoiISBFJ9x37eLL
bYE+aQcMKQSxGaDj1h385+OdAzsDoCRg88UcEU/710yougCdgMH+nNTyc97YzvdzUPDT/OK48JW7ua9OK
obC8br8F0wxVwkuTFC888YcLEwOn1eSZc9zYS+/nAiIihSZPYTkpKYkWLVoAMG/ePPr16wdAXFwcKSkpBdY
5ERERyd2oBS6mbYQrGzsZ3NqDzQLeAHy8xsGHfznwBg2i7CYVnSbbUjLLjFbDpEtNP/0b+DCAx/5wccBj4c8k
G38mZf/T4OyqfgY1L9qFuk5Vk9gAMzbbSXZb2JNuUCWydA1NfmGJi52poe/vkNYe6pXXqtciIqdDnsJyrVq1
WL16NcnJyWzevJvv1KrVq0C7aCIiIjkdMBt8MNGG0ETPlvnYF2yhZaVA/y0xZ4tGKf6DFJ9oVJqR
WeQKxv76NvAR+WIzMD4Wc80PvrLwflkKxsOWrAY0Cw2SIeqfq5t6sVSwiuxTWMzw+PaZCtVIkvPIl8/bbYx
bZMdgI7V/FxxRi5DDUREJN/yFJZvueUW7r//flWiWC+3ataN58+a8+eabvPHGGzz33HMF3UcREREE5yrwdVoJ
mZopdusfG0j2Zv9YbVQhwaSMfy/dY2ZFqoUtNP1c08hJpz3mtKpEmj7QL7UHs9oeGKRfHxbpOVuPYQPjxu
mQL59Usws6cRuuSLTy3yAVAeWeQxzu4S90QcxGRopSnsDxgwACaNWvGtm3b6Ny5Mw5ctCtW7fmo48+ol2
7dgXaQREREclpzrbQr/AKLji7io8Zm0MpuFlsgJ51fVx+hg+bBS5rlL9Ko6sUbioZbQ+t4L3lsKXYL/L16Vo7X/zt4
MrGXq5rmvv37q99Fu75JZLDR0YNjGjvIa6UDS8XESlqef6V2KRJEypXrszy5s5ctp1aoVtKlSgXlgEgtx
/+2Bn6FX5BXXi0jZsbm3uJdZnEuhSYjqVpbIAthy2sSy6+i3y9s8rBO6ucALz6p4vyDpNLGuQcMv7rVhtPLnCFh
9c/1NbNBaV4/2gRkaKSp7Ds9Xp58sknmTRpEhaLhZkZzZmBhSUlJ4/fXXiYmJKeh+ioiIyBFLdlvD2z5dVD8U/
BpW0KJOx9M0NsDMzXb2uS0kpRvEF7Mq7HtZgnKGZxe5KOd007m6H4sBGw9ZeGeVg5+3ZI6lH9bOne/RA
yIikjd52oHv9ddfZ9WqVXz++ec4Ef5Lfccgu7du3RdfLNAOioioiISHYZQ7AdFpPEOkXcmRKiSbZFvorXBsS7U
g3+tyq0R3JcRJAnz07HbjEJmAYPzY2g6zfR9JwSxRU/RIWDcjmHyXPnpisoi4gUoDz9tpg+fTojR47kLPOCh9r0
6YNTz/9NL/88uBuBdU5ERKQs2JZi2mXl6Yh2qBohyFFYvS7YmFbMu8lW85i3/v
MWGSeh7/0LndHrX9/PE2W5sllD1O8VnsDc980+2rrV8fN0rlW51NPRaRKQg5WkY9p49e6hevXqO86VKlSI1
w4cOnvVMiJlxeq9Fgb/GlqoadK/AT7unkbUUStWB014dpGTqRuyJ+PEmn7ysfxImRbtyFzkq7iF5VlHqsU1o
4O0qBSqgPeo66d1XCqLdllZtsdGuh/axAc4u2qAutpHWUSkUOSpsty0aVNmzqV4iECRNo0qTJWf
DnHit3Z1nReNMhK6MWuDBN8AVgv9vAH4QXl+QMyo0qBOhWW0Nw86PJkS2k1u4vPsOwd6YarNoXCu8
X1fZlW3isalRoga8nO7p5vrObKxv7FJRFRApRnj6Ofuihh7jllltYvnw5fr+qh8eqeedgu6jiIhIqeP
2w0O/uUj1GxiY1Csf5L+DVn7ZaufKaRa2HrbgDWYfll2nXIBXEtOJjzRL1T7IhaVpbIAfN9vZm25hb7pB5YiiX+Rr
1pbMP8Uu1LBqEZFiJU8frSYkJPDFF19gt9upUpU6ePhtonr16kyaNlIlzzjmnoPsoIiJS6izebeWgJ/Rr+KG2Ht69
MI0aUaGq4YaD1hxBuUZ0kLcuSKdWjILyyWpagIt8/bDRxqt/OvEGcj8naMK2wwY7Ugz2phus2wvTN4WGYN
eOCXKGVjQXESlW8jzRqWnTplr5WkRE5DT5bXvmViTXmvmtZ96vuItMO4xHQem+/CakCruAC1YoIc8hoEgnBpI1+x2+
6opMm6yNfaZCudaxwn2ebDvwcsPPFHBADlnSY3NvPmOGdtsoWnFrj458DRn3SEvu561BBsEREpermG5cc
eeyzPF3n66adPSdERERETKAtOEeUfCckKVAJFHFvRqWCHIFxenFWHPSrdoB9SKDrI1xcLa07jI1xd/Z67I9uMm
W7awHDTh3VUOPvjLQcA8dhqOspn0ra/55yIixU2uYXnTpk2F2A0REZGyy/q+kIr
ZYWHeahmEnuw2mb8wMy+sPWNl00ELd8kH8QRi1wBUeam01TK5r6qV2TBBPwKBmZRdRgVTqVwgSbc+t
BRERKSq5huXx48eHn4+OjqZhw4Z5vtPRo6NE6t6PhaqiPwlntPadmyQEBhjaQG2EQnhTAoiNWcpgRM2M7cSDAY+qFumiZer5dVq1Zx5513FlgHRURESjpfAJ5Z5
OKHjXaaVwqQfiQf1y8foHq05iEXpiZZFvlal2yh0ynMW96XbjDxn1BJuEnFAEFg/X4rP2628d9BSzgoN4sN8Op

56VR06XstIlKS5CksT5kyhccffxyv14thGJimiXFkFYratWvnOSy/+uqrzJw5E8MwuOyyy7jpppuYP38+o0ePxuP
x0LNnT+6//34A1q5dy4gRI0hNTSUhIYFRo0Zhs9nYsWMHQ4cOZd++fdSrV4+xY8cSFRXFoUOHeOihh9i6dSu
xsbG88sorxMXF4fV6GTFiBKtXr8blcjF27FgaNGhwkm+XiIhI/qT64OHfIli4K/Qr9699mXNlO2sIdqHL2GsZQot
8nUxYXrLbygtLnPx3MPN7eVUTL0npFtbvt7LpUOgfgDZxfl45L50oDbMWESlx8jRh5+2336Zfv3789NNPlCtXj
kmTJvHOO+9QrVo1br/99jw1tGjRIhYsWMC3337LN998w/jx41m3bh3Dhw/nzTffZq0aaxevZo5c+YAMHTo
UB5//HFmzpyJaZaZjJaZpMmDABgFGjRnH11VczY8YvgnAK6+8QkJCAtOnT+fyyy/n2WefBULDySMiIp
g+fTrDhw/n0UcfzfebJCIikl9r9ll4eqGT/t9FhYNylcjsQ247Vz89qzFL3pVzhLbhAlhzEot8BYLw3CJXXtqBcv3yAbr
X9dKudfZGu2jFBxnZRUBYRKanyFJa3bdvGTTfdRK1atWjSpAl79uyhc+fOjBgxgk8++SgZhsdeyfRSPPnkEm83
Gvn37CAQCHDp0iDp16lCrVi1sNht9+vRhxowZWN3bh4NNbA9KKOtRbHwsSTBrMlHfdrK1atWjSpAl79uyhc+fOjBgxgk8++SgZhsdeyfRSPPvkEm83
Gvn37CAQCHDp0iDp16lCrVi1sNht9+vRhxowZbN++HbfbTevWrQEYMGAAM2bMwOfzsXjxYrp3757tOMDs
2bPp06cPAL1792bu3Ln4fD5mz57NJZdcAkC7du4lTk5mx44d+XqTRRERERejuf3HVYenOtiZVLo1+rK3TDop0imb
nCQ7A4dO7e6e6n4m9U3mhczpxEUHOqebnzMoKy0Wh5ZH3ffEuK2n5XIR61lYbWw6Hvqe96/l4u2san/Zlw2
6F6tFm+NrlHSavJKZR3nlauy4iloUoT8OwIyIisFhCvxjq1KnD+vXrOe+882bdN2N2u2u53/+7//44MPP
qBHjx7s2bOHuLi48PPPx8fHs3r07xP/G4uDh2797N/v37iY6OxmazZTsOZLvGZrMRHR3Nvl37cm3r8OEDsG68OZ
HQ2NqvY2NQ+rahPmTDBhxTQ0M9hhbXSfJ61HPa+eNR4cAHRgV/W9W2q4n+97u4pKcrnGTU/s48Ws/v/PcA
NmWHw7NGHjsGy2YTQ+gnFGaAyM3HsM4OhaJCKtQLaZjE7MRr8kPHtvBa4psAowHe8mXMFzjPW70b2v6XpSx67Kx
p1ZIsISyhyYjBr1cWTJbWzfbCkxrOCt8nK+pYyRUeGNKfwUdRvCDM/tMDYgRVrjMnE0zPdQMw+CTbgXsNDTpZ+t
FXvCHaAW94u+BSSPmbs5RBjGwuxDzqLjS0Z4mxHhIFSyYD3Q17VdjvB9VjB9yLEeMiIiLjpMRcjF+zUJs8TMrk9
mhBrMQ11kVmoFz8ykvHZTwY7W+Kn1DTwjVf1wREPQpQW8kbJbHXmVh8RlXmxmqTiXJ+wVwM3m0xGqPxVqJ2O
TDQfD4XMmdz/mo5FdA3WVuqlAqCy4QQYP4Y4a9m5s4+BnrZ7TqHR+Hu2o3PLD8lQUfjdgUfsXkDeqnRMtMCT
xnhWY+b/rO6QBhS+pVpG3PNB8lmY9cNZtS6/79+TTNyg8O4rvOY9lIx9Wp0+z3oYOvOX1udz4PC7fR2gh2QJDdeJ
dTutWdi4uq6u1PUZJ4ZbjqgnUsn4cg+ASd9LT1ddpDz99wPSO6wT9vqd1fpi8aRR1ZYXktpddwjnlvXZ2eTe9bIt

UREZEikmtY7tevHx6Ph6+++gqr1crLL79M69atAXj99df5+uuvueWWW+jXr18hdVVEROTU+ALw2vJQFXnhL
hv3zY5g0e7Qr0KLYRI0Db7faA+ff/9ZbvrU92kFZBERkTLouDsFDhw4kIEDB+Y4fvvttzNkyBAqVqxYYB0TERE5
3b7faGd3Wua85IygbDVMxnZJ5+HfIvAFQxXlqxt7uaaJr0j6KSIiIkUv1wW+jqdq1aoKyiIiUqL4g/DRX6EScZXII
NWisuyrW99H5xoB7m0T2i4osaaPIUcei4iISNl03MqyiIhIaTFto43tqaHPiG9s5qVNfIC7fonANOGWFl4Armzs
o2fd0LBrrYIsIiJStiksi4hIqRY04a0l8OIiFwCVXEEuaeDDaYWpl6RiENoiKEN57YgoIiIiKCyLiEgJFDRD2z4dzROAh
Tut1C4XpE6MybI9Vv63ysGyPQAGNovJwwkenNbQ+RH6LSgiIiK50J8JIiJSokzZYOeFxU5axwUY0sZDk9jQ3G
N/EO6bHcHiI4t2xdhNDmfZ/qIWdJDnOqXTNDZ4zOuKiIiIZKWwLCIiJcauVIOxS5x4gwaLdtu4boaVPvV9DGnt
5bN19nBQBsJB2Wk1uaqFwY1npBJtz+3KIiIiItkpLIuISInx6p9O3IFQCLYaJgHT4Nv/HMzeZueQN3S8QfkAfRv4
WLPPSq2YIJc18tGkdjRJSUXZcxERESlpFJZFRKREWLLbyk9bQqXhbnV83H6mh5eWufh9hy0clKNsJi90TqdOO
RPQHskiIiJy8k5qn2UREZHCdNgLoxeHlql2WU3ubeOhTjmTVxLTGX1uOpVcQZxWk1Ed3UeCsoiIiMipUWVZ
RESKNU8AHpwbweZDoSWsb2nhpUpkKBAbBlxUx8/5tfyk+yHGUZQ9FRERkdJElWURESm2AkF4fL6LZXtCn+
2eX8vHdU29Oc6zWRSURURE5PRSWBYRkWLJNGHcMieztobmKbeJ8/PMOW6s+s0lIiIihUB/coiIlBG/brUxY1
PxnH2zfr+Fedut7Hdn7ov88RoHE9aHysX1ywcYl5iO01pUPRQREZGypnj+1SQiIqfVlkMGQ3+LAKCCM42zqw
WKpB8HPPDV3w5+32Hj1nbQuRKs2Wfhph8jCZihoBwfEcQThIOe0Oe5VSKDvHZ+OuU0zFpEREQKkcKyiEgZ
sG5/Zkl26gZ7oYfloAkfrHbw8VoH6f5QKH74J/i4u4XXlzvDQRlgT3rmoKdyDpPXzk8PL+glIiIiUlgUlkVEyoAthz
MD6JxtNlK8EH2aKrX/HrDwzEIXFgPGdUmnoit7sPUFYNQCFzM227MfD8K9syNIOhKOL6rto0H5IJsPW4iym
1RwmvSu56NmjIKyiIiIFD6FZRGRMmBrlrDsDRr8vNVOvwa+U77unG1WHpsfQdqRavErfzoZ1dEdfn5nqsGo
BS6W7A79uqkdE+SRdm6W7bby/l/OcFCOsJkMTfAQ61IwFhERkeJBC3yJiJQBWw5l/3E/beOpf1Y6dYONh+Z
mBmWAHzbaWbLbitsP761ycNn3UeGg3LJygA+6pdKhaoBbWnhpVjnzWtc28Sooi4iISLGisCwiUgZsSQkFWo
NQIF22x8aOFON4LwFgV6rB/bMjeGeVAzNLlp233cpzi1yYGETYTIa1c+Oyhk544g8Xfb+N4u1VTjyBUBu96vl4
84I0KjhDr7db4eXuEOsK0qB8gGuOsXeyiIiISFHSMGwRkVLuoCdzZenudfzhucNTNti5q1XuIdU0Q3ONF++28
dsOG9WjgvSu72f1XgvD5kUQMA2cVpM3zk+jZVyQNB/833IXu9MyP4dtVCHA0AQPZ8XnXFCsSWWY1i8Vq
wHGiXO7iIiISKFSWBYRKeWyzlfuWtvPpkMW1u23Mn6tgx51/dQvHzzm6777z8bi3Zm/Jl5c4iLF5+H15U7cA
QOLYfLcuem0jAu9/uomPn7eYmdNspU65QLc2sLLRbX9WI8zhsmm8U0iIiJSTCksi4iUcllXwq51ZIGtm3+KxBc
MLb71/kVp2Czg8cPEf+ysTbZQr1yQD/4KjZmOsZsc9hmk+g3GLnUBoeHcw9p5SKyZWTG2WeCdC9PYdMhC
owrB44ZkERERkeJOYVlEpJTLWlmuGR3EZYNrmvgYv9bBX/us3Ds7gtoxQX7bCbtSXDle/0RHNwt3Wvn6n9B
eU1F2k2fPSadTjZxDq102aBJ77Eq1iIiISEmisCwiUsL4ArA8ycq/Byzscxu0qxqgfZVArvN+Nx8Jy1UjQ0EZ4PYzP
czdbmXzISsLd9lYuCvzfIfFxBsMXeyi2j7Oq+nn7Kp+DnoNUrwGD7R1U7ecVq4WERGR0k1hWUSkBHH74ZafI
lm33xo+9tGa0P7FlzXy0ru+j0Neg9f+dLJ0j5WH2nrCleXaMZkVX5cNxpzr5qVlTnamWkh2G1QvZ3B1o3Quru
dnR4rB9hQL7aoGwuc/d64bERERkbJCYVlEpBg64Db4bS3s3h9aubpjNT81ok3eX+3IFpQzbDls4aVlLt5Y4SRo
gu9IZXj0IhcZEblWTPbh0Y0qBnmra3r467i4GJKS/ADULmdSu1zOYdYiIiliZYXCsohIIfv+PxvL9lhpGhukdVyAhh
WC2YZQz9tu5ckFLg54AEJziJ1Wk3vbePhkbWjecLPYAGO7pOO0mnz3n51v/nGwNcUS3tc4Q6o/8+va5TSX
WERERCSvFJZFRArRkt1WnlwQAcC3/4WOVY0Mcn4tP+UdJhsOWvhpiz3H6zwBgxeWhIKzxTAZ3t5NfGRo3v
C1TX1c3cTHol1Wpmywk+43uLWFh/dXO/ltR+aP+doxCssiliIieaWwLCJyEgJBeGGJkz922niyo5uz4k88ZNnth
+cW5VxteleahS/+dmQ7FmkzeeYCg2ZRKSzZbeWJP1wEzFCVeOAZvhwrTlsMOLtagLOrZfbj/rPc/LErCv+RIdl
HD8MWERERkdxpF0wRkXwKmvDMIhff/OtgR6qFEb+7OOTNfs6S3VY+XuNg08HMH7PvrXaE9zy+r42bib1T
GNrWTdt4PxYjVCWOdQXpVN3PZz1TubQpVI4w6VHXz9gu6UTZTOqVC3B7S0+e+lm7nMnVjUMdq+gMUiN
KK1iLiIiI5JUqyyIi+WCaMG6pk+/+yxwqnZRu4cUlLp4+x40nAK/96eTL9aFK8WvLnZxRMYDbb7D1cKjC2yw2
wJWNfdgsULecj4GNfaT6wGoQ3trpaJ1rBJg5IAW7Baz5+Jjz7lZe6pQLckbFIPac64KJiIiISC4UlkVEjsMXgK0pFu
qVCy3CNXEtfHUkCNcpF6BqpMnCXTamb7Jz0GOw6ZCFHanZ0+z6LKtX2ywmIzu4sR0VeKNyTlPOIbcgfTxWC
/Rt4M//C0VERETKOIVlEZHjGP67i1+32WlXxc/NLbw8Nid0PNYV5I3z07FZYOAPURz0GszfmfkjtUWlAPe09rB
wl5Wlu23EuoLUjgnSs66fRhU1d1hERESkuFNYFhHJxeJdVn7dFir5Lt5tY/Hu0I9MA5OnOrqpemQO8KiO6Tyz

yEWENbSIVvuq/vAw64QqAcCbWxMiIiIiUkwpLIuIHINphuYbQ2irpqCZuV/xjc292Vad7lQjwIz+qYXeRxERERE
pOArLIiLH8MtWG2uSQ3ONb2zmpWZ0kLdXOulQy8LtZ6pSLCIiIlLaKSyLiBwlxQdvrAhVlcs7g1zf1Eu0A/rU9x
MfH0NSUhF3UEREREQKnPZZFhHJwhOAh+ZGhPdDHtQ8FJQBDOM4LxQRERGRUkWVZREp8/akGSzZbSXN
b/DbdhtLjizk1aWGj4Fn+Iq4dyIiIiJSFBSWRaTM+mWrjc/X2VmRZMUke9m4dZyf587NuR+yiIiIiJQNCssiUiZ9t
s7Oy8tcx3wuoYqfFzun49JPSBEREZEyS38KikiZYprwwV8O3loZWsAr2m7Sv6GPC2r5qBZl4rCalHMUcSdFRER
EpMgpLItImeH2w5jFLr7faAegojPIGxekc0bFYBH3TERERESKG4VlESn1dqQYLNtj5bN1Dv45ENo7OT4iFJTrlV
dQFhEREZGcFJZFpNQ64IGRv0ewYFf2H3VnxYcW76ocYRZRz0REREESkuFNYFpFS6b+DFu6fE8H2lMzlrCNtJg
Mbe7n9TK9WuRYRERGR41JYFpFSxReAL/628+5qJ+n+0HZQF9X2cW1TL40rBhWSRURERCRPFJZFpFQIBGH
WVhv/W+Vg8yFr+PidLT0Mau7FMI7zYhERERGRoygsi0iJt2CnlTGLXWzLMuS6XrkADyd4aFc1UIQ9ExEREZG
SSmFZREos04R3l8Fz8yIImqHScQVnkBubebmysU9DrkVEREТkpCksi0iJc8BtMG+HlR8325m/E8DAaTW5p5
WH/g19uPSTTUREREROkf6kFJESwzThm3/tvLTUiTeYOQm5SmSQcV3SaRKrPZNFRERE5PRQWBaREiHdD6
MXuZi2yR4+Vs5h0r2Bwc1N0rRnsoiIiIicVgrLIlKsmSb8us3GuKVOdqeFJiHHRQQZ2cFNh6oBqlWJISlJQVlERE
RETi+FZREptoImPD7fxYzNmdXkhCp+nj3HTSVVkkVEREESkACksi0ixNWG9PRyUyztM7mntoW8DHxbtmSwiIi
IiBUxhWUSKzO40g0AQqkfnrBJvOGDhteVOAKpFBfmkexoVXaomi4iIiEjhUFgWkULn9sO7qx18utZB0IQ+9X3
c1cobXqTrsBcem+/CEzAwMHmqo1tBWUREREQKlcKyiBSa/W6D6ZtsfPm3gx2plvDxb/9z8NMWO/0a+OhQ
1c+LS1xsP/L8jc28tIkPFFWXRURERKSMUlgWkQJnmqFK8gd/OfBn2R+5eaUAsU6T33bYSPcbfPG3gy/+doSfT
6zp47YzvUXRZRERERERep4xSWRaTAvb3Swft/OcNf14oOck1TL/0b+LBaYPEuKx+ucbBoV+hHktUwubuVh+ua
+jC0mJeIiIiIFAGFZREpMP4gvLPKwQdHgnKVyCBPdXRzVnwgWwhuVzVAu6rp/LPfwq/bbJxTzU+LysEi6rWIiIi
IiMKyiJwibwC2HLaw9bCF3Wk6G+9INYqLBk+7g+412tqeE5h7HRQR5u2satWJyX6irUcUgjSpq2LWIiIiIFD2FZR
E5IX8Q1iVb2HTIQnmnSaQNluy2Mme7jX8PWAiaxxornTnsukZ0kFfPO35QFhEREREpThSWRSQH04SVey0s
2mXjzyQrq/ZaSffnbfKwxTDD4TkuIshNzb30a+DDYS3IHouIiIiInF4KyyKSjT8IYxY7mbLBccJzG1cMcHY1P/XLB
6kTE6RqlEmsyyQuLoZtuw7jsqIFukRERESkRFJYFilD0v2h4dNbDlvoWC1A/fLZF9FK88Gjv0fw+47MHw11yw
VoExegdXyAJhWDpPvhgMegfvkg1aOPPazaYkCEfrqIiIiISAmmP2dFyoB0H7ywxMmUf+14j+xz/H+GyWWNf
Nze0kM5B7j9cO/sCP5MCv1YaFIxwNgu6VSN0jxjEREREESl7LIXZ2Ouvv06vXr3o1asXL7zwAgDz58+nT58+dOv
WjZdffjJl87tq1axkwYADdu3dnxIgR+P1+AHbs2ME111xDjx49uPPOO0lNTQXg0KFD3HbbbbfTs2ZNrrrrmGpKQ
kALxeL0OHDqVnz57079+fDRs2FOYtixS51Xst3hIMyQMA0+Gq9g0u/i2LGJhuPzssMyh2r+fnfhWkK
yiIiIiJSZhVaWJ4/fz7z5s5s1j8uTJTJkyhb/++e4cOH8+abzDBg6dCiPP/44M2fDOxDRNJky
YAMCoUaO4+uqrmTFjBi1atODNN98E4JVXXXiEhIYHp06dz+eRx8+trFsW
KTLeAMzYZOPWnyK48ccccoNuwPHW8d52dclzQ+6pZK2/jQB1D7PRZGzo/gtyNDr9tV8TOuSzpR9qLqvYiIiIhI0
Su0sBwXF8Ye7Y8WnwOBzaNZMz/9mzmxBDzADq0xP5tzt3bcbvdn69e4YNGN0iBvAAcjPKShoNAdi4wWnFnbz7cgHuEW9f
4fixcvpnv37tmOA8yePfPZEs+ffA0Lt3b+bONt27NlRWLctUUdS/fDBXw4em+/it
eUOXl7mpNeUKEbOz6wU2yxwTysP/+uaTmLNAC0qB3m7azrPnJNORWfmvOWmsaGh11q5WkRERETKukK
bs9yoUaPJs+Kl0mDzXpmdiDwbDPiisC3cvXiNLA/GSjvJtK4+ao+CNNm3bUXlrDIiL6O0+uzdjLZ1dE9+jwSuGfDl5wRPf5vMPxJf1ZMYs0mp8+TmkVlt0HFHCh/8fDBnE+zFUNpJaesFc/S/NXJI0rTM+wmYp
ZjX2rVrF9WrVy/Q+xDTs/d1+yTLGkVWKwh6whBTZsD9mX3+IeDX1Bo0aVKe0PAvQOpaZxuU2ZfGrAmSjuLFxrsKaE3QwLBhtSPVn9VRFR9iVSzRMbYXjZ5wfvJFJWb8OmfChCvdGgIM5
9q1a2e7Vl7t25dCMJj3RY7i4mJISjqc5/NPVWlur6nyzoZAbbwF8fF1tsNDG03sFVbpLfqvvoWzhgh2N9YDP98cgtqbIYh5+V9BmZXqJW3fq0kHevkUT2+TR6hXRTT8GCb+fiM39FDk
OEq8OH98PhPLZ3OhW397Kktlea762w29O9qb3i1lZht1ea762w29O9lcz2SvO9FXZ7peneLBbjuMXRQgvLO
3fu5O677+bll1+mY8eOALRq1YqNNGzezfv15ZI8a+efNmatasyffff88dd99NkMmmZmNzTDdxxwzjxj+I6JGKcTM
dOm0adPH6ZMmUKXLl0ASExMZMqUKdxxxx0888wz1mqUkx8swXFLQ+SLExMpqqL9xNV/UhjBTXomzITYbwKfm6s1tTsL+kXn1fm/N6NnwuxiEpwlPPixHtMKfk
cVO2DXbrY8bMiI9oMPOuT8Np5VWUVkQ7YQbm8cw8Dq0/

l7lYMYmO55AZlI2MDHJnpzjIoK8dUE6dTP2SzYgqlDXuBcRERERKZ0KLSy///77eDwexowZEz525ZVXMmbM
GAYPHozH4yExMZEePXoAMHbsWEaOHElKSgrNmzfn+uuvB+CJJ55g2LBhvPXWW1SrVo2XXnoJgHvvvZdhw
4bRq1cvYmJiGDt2LADXXXcdjz/+OL169cLhcIS3rBIpTg56YPFuGwt3WZm+0Y47kHN6wdFBuWpkkLe7plEzRt
s7iYiIiIicboUWlkeOHMnIkSOP+dy3336b41hYjxYPFCBt99+O8dxp9PJ888/fxI9Fil4
QRO+Xm/n/5Y7s1WRAc6t7qdZbACArHHYYkCU3aRnXT+xLgVlEREREZGCUOgLfIIIyAG3wYj5Lhbuyvzf0Gk1a
RMX4MbmXhKqBIqwdyIiIiIiZZvCskgh2XDAQrr//9u708CYzr4N4NdkXy2htiRiKaJUi5baSqWaJWUCVajVL
UTqtPyEJCxVYUrb7WUEtDWqqhiFbU0thiCxGJIIQsksgqk8zc74c8c0iqTJIzk8y4fl/IbNf/zP2YPP2P2YDWtdXILw
SmRFjj0v3iCxrXt1Xjy/YP0bG+Cpa8xjERERERUaVjs0ykY7mTFwMpzlthxrfgSTt0ci2BhAalRfsu5EP5vPIQdr29M
RERERFRlsFkm0qGYdBN8ccQa9/IenaL6S3C150fwpxbk4mIIiIqhQ2y0QyEAJIzlRxLTAa3VVS
HhQ3CjnFhWfuKtdnSLUtxUISyjehNykugqLuuWzUSYiiiqoLYLBOVUq+AvEPTPCSgwoWpsCuOHP8bsNx7i9Xoq6TYTBWBlBmRXxpsmIiIiqLnHptl0pmbWQqM
OWSDtPwnX/jYVCEw8MVCaWsyERERERFRVcFmmWSTUwgculW8SMtHtHpio1zdzQqCbB5vgj8dHln8a14cm6iIiIiIjIeLBZJ6KnGQwWOJZ
kiS6lAllKBLTEWyC8q3oe6dS0Vvu6SD7sadlh0pBD7bjxqlF9yUOHDFoWV9baJiIiIihkx2aZEHnPFFuuWCDynil
U4p8HGH/csgCfv6KEmQnwQk0gsPNDDHixEMvOWiJLqcDsTg9hxrNbExERERGREWGz/BwqUAhxQQUHXgt+DvNMOxr
ga4wFjt998mJQz0aNR0eonMD1T/ua1dHhU3v5un6rRIRERERERVUKNsvOfGfziQdRAuG3zM1kTEREREdFzic2ySpZs
mxjJuDeqBDvNircizVUMDcBLE3BM1kTERERERFRVcFmmWSTUwgculW8SMtHtHpio1zdzQqCbB5vgj8dHln8a14cm6iIiIiIjIeLBZJ6KnGQwWOJZ
8QRcRERERERHAZtloqAVg8r9e+FKaCbZftcCft83+0SBbmwn0dC7Ca3WLUNdGoHE1NV6wYZNMRERERET0
ODbLBkoIYMU5C/ydaIBT/ydAiRl2SGvCHC0FbAxF7j6v0s9adSyUqOrYxE611ehc4MiWHOuExERERERPRXbJGOVnKf
ApiuW//ureOtxYs6jrcimCgG3hkXo17gQHeqpYMqzVRMRERERRWmNzbKBqmsjMLZNARJyLVHNRAkbc4Fb
2SZIyTNB+zpFGNKiEHW5ezUREREREVG5sFk2UAoFMKq1Ei+8YERERERERGhTvnEhEREREREZXCZ
pmIiIiIioFDbLRERERERERERKWwWSYiIiIiIiIqhc0yERERERERRUSlslomIiIiIiiIhIiIiEoxq+w3UNWZmC
CwTERERERERRlcJmmYiIiIiIiIiIiKgUNstERERERERERERERFRKWyWiYiIiIiIiIiIiIiIiIiIiIiEoxq+w3UNWZmC

j08pyKMOY81sa8qpal7zxjrk3feayNeVUtS995xlybvvNYm2HmGXNt+s4zltqe9boKIYTQSTIRERERERGRgeJu2
EREREREREESlsFkmIiIiIiIiKoXNMhEREREREVEpbJaJiIiIiIISmGzTEREREREREFQKm2UiIiIiIiKiUtgsExEREREREZXC
ZpmIiIiIiIioFDbLREREREREKWwWSYiIiIiIqhc0yERERkYHLyMgAAAghKvmdEBEZDzbL5ZCTkwNAP19I+szSd
54x16bvPNYmjxs3buDMmTNGmcfamFfVsvSdZ4y1aV73r7/+wtdff43CwkIoFAqdZD3OGKdlZWTpO8+Ya9N
3Hmsz3LyyYrOsJc3M27NnDzZs2ACVSqWzLyR9Zuk7z5hr03cea5PfihUrsG/fPuTk5BhdHmtjXlXL0neeMdW
mGSM1r/vzzz/j4sWLOHjwYIn7dcWYpmVlZuk7z5hr03ceazPcvLJis/wMpb+QQkJCEBkZKf0CYqhZ+s4z5tr0n
cfa5CeEQExMDCIjI3H37l388ccfRpPH2phXlX1bL0nWeMtRUUFEj/T0xMRH5+PgYOHIiIiAgkJydDoVDopGE2x
mlZGVn6zjPm2vSdx9oM6N6+82Cw/g0qlkv5/9epVmJubo3Xr1jh+/DgyMzMNNkvfecZcm77zWJs8IIiEBMTg
6KiIigUCiQlJWHQoEHo2LEjoqOjcevWLQDybaHRZx5rM8za9J3H2gyvtlOnTmHkyJEICAjArl27oFaroVQq4eb
mhs6dOO8PGxga7d+8GANm20BjrtNR3lr7zjCLk2feexNsOsTS4KUZNq0CU5OTnBzc0PHjh0RGR
mJ9PR02NraIw8Hf8GAN5m20BjrtNR3lr7zjLk2feexNsOsTS4KUZXeTRVy4sQJbNq0CU5OTnBzc0PHjh0RGR
mJ9PR02NraIjw8HB07dkTfvn0NKkvfecZcm77zWJs8WcnJyZgwYQJsbGxQvXp1uLi44NNPP0VaWhrq1q2Lp
KQkbN++HS+++CI8PT0NKo+1GWZt+s5jbYZZ24MHDzBmzBh4enrCwcEB27ZtQ6NGjfDFF19Ij9m9m7dy8OHTq
EkSNHolWrVhBClLtpNuZpydoMszZ957E2w6xNbtyy/ATJyclYvHgx+vXrh0vXXh0aNGmHHjh348ccf8cYbb6B3797o
3r07nJ2dceHCBdy8eRNA+X8B0WcPXmaPcPNYmX23Xr19Hhw4dEBwcjMmTJ+Phw4eYP38+XnzxRdjb26NFi
xZo3rw54uPjceHCBYPKY22GWZu+81ibYdYWHx+PRo0aoV+/fujSpoaoV+/fujSpQu8vb0RHByM+Ph46TGvvvoqXFxc8Ntv
vwGo2NZlY56WrM0wa9N3N3HmszzNrkxmb5Cc6dO4O4dmzZqhd+/eGDx4MDw4MDw9PfHjjz/eGDx4MDw9PfHjjz8iMTFRekyXLl2gVqtx9
OhRAOX/QtjnFmsz3DzWJl9tFy5gtLVr1wAALi4uPLyQnR0E6cOE95s0334SFhQUOjwIpVpMHmszTBr2bs/OZ85d5fGhs3mg5ERRkOkrQJTQE
zRBqkxub5cdoBoHmzZvj5MmTSE9Ph4E9Ph4WFBdq1a4c333wTK1eulB7B7bqlUrODs749KSl9LWrqqaZcy1CSGMdlq
qVCqjrU2fWZocob/vv/8+UlJScnOnSJZmaFBgwYYYMmQItm7dkKj3H0dERrVq1QsOGDctdl77yNWHHr+sh6/
PJk+pxv+qgvOTkZQMnxxhqq4wsfXGGNOdneLwxUaiUqFu3Ltzd3bFhwgMLp9smTjJyM7OxuXL1+Wbnb
g4+ODVq1alb+Xdf2pByOJxXL09BHFscTw/0MVMa6SWXk6cJz3SyXvn6r5hcMFxcXdOvWDevXrFxXr5fuHzJkCDI
yMqQDzwHA3d0d48ePh4uLzOz0tPTS/ytyyyg+NgmfeXduXMHMHwKMvI13XFhMTg4iICCL13n6nHcXLlzA3Ll
zAQCmpqZZGVVvpE3TpMuvy5cvYvXs3srsKy/vHLZI0aNeDm5obNmzdLea1bt4aVlRWk5ObN
CwsnpmXnJyM48ePl6hL86/ceZcvX8bmzZtL7Hqpy9quXAXXd4ZCnRL1yU9RQ1Gv/4rFx5Lkyr/Yj+ZvXx=Hqpy9quXLmCn3/+GWlpaTqvLSUlBX5+fgCKl39dZgHFn7fJkyf
jwIEDUpYu8y5duoSNGzciJiZGuk1XeTExMRg+fDjGjx8PoOR4oovaoqOjERwcjOvXr//jPl18BkJCQkrsfaKrLE2e
vsaTK1euwM/PT/q8aZ6vVqsBAGPHjkVUVBRONToFALCwsICrqyssLS2l17C3t0fdunWfZNo4nhjee6HM
sATieGOp4os91E01t+hxP9OW5a5Y1M+Po0aNYtWoVsrOz//GFZGJigr59+/Ls2bO4cOECFAoFrK2tWdnBxs
bG+m16tSpAycnp6fmZWZmIiAgAOPHj8eSJUtw+vRpAEBRUZHsWZq82bNNNY/Lkydi0aZM0KjVap3kpaenw
83NDbGxsTA1NZUaZl1kZWRkwM/PD35+fjAxMZFyhBA6ycvOzsacOXXPw6aefwt/ffHxcvXpTukzvvvvvYMHmD
NnDiZOnIizZ88iPz9fZ1kAkJWVhTlz5mDcuHFYtmyHylSqXSW1TpkzBihUrpJU8XXwGhBD45ptv8NVXX+
H48eMIDAzEzp07ATy61IlqSV69+NtLQ0bNu2DQqFqq7bt260heJZhnTxoYNGxqeqeFHiRlcLOJfD6y1UpPWVWhRrfzdLKJ8
+9BaD4s62Laan5zGmWy4ULFyIsLEwntanVasyfPx8+IB17DVjEGrOC+HNEJSUWlCN2wZ3kcQ7UdmSmhr0Evn5nCZVlOKC8
wN4tGJrbm4OoHj5nD59Ovz9/XHixIxqFmzplavr8rHxxHDHE32NJQDHE0MfT/SxbQ5XX2OJ/pmV
tlvoLLs3r0bV65cQfv27fH222//YvZZCWVlhwlABmDVrFr777zjs378fDx48wGJWVDZly5s6d0dHRwRwQFBeHgwY
NYuXIlNm7cCDOzR5NeiwACAwMRMP369Ylb06pyYyZYf868yZ5++Xy5YUyGgQBAB0Rrkk2bepKZBzKSYtn3f42IRbfedMrKPwxGZm5xeNHFCQTFFhZQ4PNsZqm4J4FjaRTlMVl0OLB6z9ImuwzzAHrG7utSRFAHsBA45Q8WstALbyZQE1zzLAzZ1AGBxXNkjw/Jiav59h+rP9HSw7hjHT5BQy7sHeeKSPtBZsmG/MX48O3Ea3CAO6S3AqWyDRXDQnAXVIsFtQyMAVm3XNjRiEU1NtEm1yaRQ+NdhUqyLLTy7JPbvu
HTsGa2trnWRpLFu2DGq1Gmz/cubNmzcPNWvRFBQEP744w9MnToVhw4dKr
ElRK4slUqFjIwMrF69Go6Ojjh+/DimTJKcNzc3aWVy48aNMDc3x7h4hx4+Dr64vY2FgcP35cOvuiENqfPVaIUiEtL
Q0RERGoU6cOwsPD4enpCRMTE2m+yZWXmpqK1NRU7Ny5EwqFAvv378fu3bu3bsxZMgQ6RdYOWtLTU1FSk

oKdu3aBaB4d0/NjxeaFXe58jSPS0lJQZMmTeDj44OjR4+WGCflrO3KlSvIy8vDvHnzkJCQgKVLl+LVV19Fp06dp
GVOzjwhBG7evImvv/4azZs3R15enjQtNTXKlTd//nxUq1YNISEhiIiIwN69ewGgxGdbztqKioqQmZmJr7/+Gq6
urli4cCHs7OwAyL+cZGRk4O7duwgODoa9vT0uXryIwYMH4z//+Q+aNWsme21qtRqZmZk6GU80t8fHx8PCw
gLOzs7IysrC3bt3MWTIEFy6dAnR0dHSrtS///47Nm7cCCcnJyxevBh5eXn4888/kZKSgiVLlsDBwUGrmjTS0tK
MdjzR0Nd4EhMTg/z8fL2NJ7du3dLLWCKEQFBQEOzt7Y1yPMnMzMS9e/f0Op7oa/1ErVYjNTVVL+smQPF4k
pKSorfxRO/EcyInJ0f6f0pKihgzZoyYM2eOWLhwobh586Z03759+8SwYcPEvHnzhBBCfPvtt2LmzJlizJgxIjExsU
xZt2/fFn379pVuP3PmjBg3bpy4f/++EEKI33//vcJZj+fdunVLuLu7S7d//PHHIiwsTDx48EAIIcRvv/0mW22P17R
lyxbRsWNHERoaKt2+d+9eWWrLzs4WQghx79494e/vL3bs2CFGjBghg8fLqZOnSp27twpW22P15ebmyvGj
h0rbt26JYQQ4sSJE+L8+fMiJSVFtjxNbZGRkSIuLk4IUTw9Z8yYIQoLC4VKpRJCyLNMlq7tv//9rzh69KgQQggfH
x8REhIiLZdyLifJycnivffeEwUFBdJ9HTt2FEFBQUIIIcLCwiqcFRkZKU6fPi2EEOL+/fvinXfekWoRQojp06eLcePG
CSGE6Nevn5g2bZo0X+/cuSMOHTpUYgzQNq+wsFAIIcS5c+fEwoULxdatW0VgYKA4f/68EEKIu3fvivfff79CeZ
GRkeLUqVNCCCFu3rwpBgwYIJRKpRBCiC1btoj//ve/QgghkpKSKpxVOi8hIUFMnjxZCCHE4cOHRYsWLcQ333
wjtm3bJoSo+LTUTMeioiLptpUrV4qLFy+KYcOGiWXLlgkhhEhNTRX9+/eXtbaMjAwxadIkMX78eDF9+nSxdu1
aMXLkSPHVV1/JUtvj9QlRvFz27NlT5OXliRMnTohPPvlEfPPNN2L79u2y5D1em2bcEKJ4emqmY1FRkbh3755
sy4mmtoyMDOHp6SmCg4PFzz//LNq1ayfmzZsn1q1bJ4QQFZ53j9eWmJgo2rZtK/Lz84UQxd8L77zzjpg6darI
y8ur0HRUq9VCCCHi4uKk5yclJYl3331XZ+OJUqkUQ4cOFVu3bhVKpVIkJSWJbdu2iejoaLFo0SKxYMEC6bEzZ8
4Uhw8fLvH8x+e1trVp7Ox+e1trVp3ltsbKwYOHCgNI7JOZ48Ke/q1atiypQpQgh5x5MnzTfNdFm1apXs48njeTdu3BBCFH/
PTZ48Wfbx5N+WSTc3N52MJU+ab3l5edL9co8nT6rv3r17Yvjw4bKPJ0+qLT4+XrRr104n44nGsWPHxMGDB
0VGRoZ48OCBTtdPNFlpaWlCCN2umzyel52dLe7cuaPz9ZPKZPTXWdbspmJmZoa2bduiV69eyMzMxJkzZ+Di
4oKwsDA0b94cnp6eePjwIfz9/aXL1WgUFhZKv2v4YYNG2L//v3MHCBA/Dz88OuXbuwYcMG
dO3aVdpFqKJ5Hh4eGDNmDN544w0oFAqFTp16oQjR45g06ZNCA4ORpcuXWSpzcPDA15eXnjzzTfx2WefITY2Fj/99B
P27duHbdu2YdOmTeWeb7m5uZg2bRoyMzPxwgsvoEGDBhg3bhyWLl0KtVotHfudk5MDd3d3rF69GtWqV
YOzszOAsv+KWTrPycuc96TaPv74Y5ibm8Pg4ICRI0fCwsIclyDQuvWrcuc96TaPv74Y5ibm8PS0hI1atR
AYGAghBDw8/ODSqVCTEyMtBWqorU1aNAAnp6e0vHhu3fvBgDY2NhgxowZWLt2LerXr48GDRpUuDZnZ2f
069cPL730EpYuXYrOnTujRYsW6NSpEzp16oRZs2ahsLAQrq6ustX26aejfIjw8HL///ru0q1heXh569OiBtWvXo
mbNmrItJ5q87777DmlpaahXrx66du2KoqIieHt7S1sZ6tevX+5pmZGRgTp16sDZ2RnvvvfeeNK327t2L7du3Y9
OmTQCKt9rExsbipZdekq220aNH4/bt2wgPD8fu3bsxe/Zs1KhXr9YylYsQIfP/9
97h79y5GjhyJvXv3ol27dggKCsK6desqNN+A4rFnxIgHR6NevHwYOHAhLS0uMHz8etWvXls44Ldd4AgB//fUXJ
kyYgHfeeQcfffQRXn75Zem+I0eOYM+ePXjnnXfw9ttvl3ie5vCVnbDo0SPr++Pbbb9GxYY0dZx5PSeR4eHrCwsMCKFStw9uxZbNiwAYB848mTTpYN+mTpuPEiRNhYmKC2rVryzaWPMy01
38lyjyf/Vl94eDguXryI0NBQ2caTf8v6/PPPMPPoVKp8PHHH8s6njx48ABffvklkl8vLy4OzsjOzsbMyePRtLliyBmZkZAg
MDAcgznpTOys7Nxcx5SJE2FlZSVt1ZVr3eTfavviiy9gbm4OTe3t7VKtWTfbxpNLpqSnXK80vSCkpKeKDDz4Q27Zt
E2fOnBG+vr5i2rRpJR67ZcsWMXv2bHHHu3Ll/vI42v9r+W9asWbOkX1E1AgICxJIlS574OmX9hbh0no+Pj5g4ca
L0mMd/uZo2bZrw8/OTrZfZX18xfvx4IYQoaGh0haGzp07i5YtW4qrV69Kz5WjtlmzZglvb2+RlZUlpkyZIh48e
CBtjfriiy/Et99+W+banzVX4moqCjRv3//EvNs2rRp0q9q9lZcl72rTUzDchin/xGz16tEhISHji68hVmx
BCXLp0SUyfPl16jq+v7z+W+W2Yp+Bv773/+KhIQE8fbbb4spU6i/idjYWDFlyhSxdOnSctWmce3aNSNSePctWmce3aNTFr1iwh
RPF0+/7778WYMWNEUlKS6NWrl7h27Zr02Dlz5oitW7eWO+tJD8eO3rmIiliStxe0dp++OE
H8cknn5R4zJAhQ0RkZKQQpTYcv/4ltqK5H388cfS/Y9/noOCgsTChQulv+Wsbd68eeLUqVNi/fr1ol/OnTqJXr
14lnivHfNMsJ0IIcfbsWSHEo+nn5+cnfvzxR1nzVq9eLb8y8vERUVJR46623xIYNG6THlH5f8YyvU/i5zniv7ltZ6ncbtmgNZ7VgWyU8+I1/7K2UFVyC8tQemtjGPw8ATEIzOUm9Fm8zr1wHp3KB3k5xfx7VTNkIlOZvEkEK7kBHJ3tsbIzoKA9hYCwLh/xIwS1V1b8OWzr2NJKPPEWNjw5qfxm1nm+kVIvdI5lB8wgDdd9a9AU5ZaxNPETmOFYefu9GyaQxR/l1w/CDKO+Xsc46TR2Th7P0U+JzUkrxLW9gjsg7vVaxOHD9uVT3z3fvHIhl2NHXiiKGdDg1h+RXX3kWPS6ZR+2Yr5GZTqbPHHtkW1sTtSGsn8OUw0uXiVRl/LKyUW14vH8VA8vTO0lD1Uiv8OH9/j1/mtwI4JMmAAQvxxbT/ODYhBdmT9ggGVuxaZSU51eepCazJ6qwyaDZgMZwXvN96XS77fhkaSEc/hrYLSt9pRXw6XRXkppC6I6HJtjLzm/SnpUwjOa9P6NyHDrx3Hph0vEGBobCbUW80BqVqEu8ybAjs3SIFyaIHHtPD0ep1PFAiJ3A3P8oUgFtQkHLIhoaDhTfd4YFABhSdw3aKLWsbJFigAZUh56GmRbv3eU4T9wt4fJoQ1RoJPB9L+XXROW2WmakO7WasY4bz7EMWWdr1JqTmEUVoAAdNuudsAq1FXPMx+tlGCHVYmk2h00dM0+AI1/nAPGwq6rbmuWpUd2oRR6cpk7zWJHPYh66sBHhnCNyAoY7+WTdnGBK9doXNl/Pp3nO+JjNrTTSq0r8m89lpxHBNzCZ5oeROf7qp+KSYdpe5bthmp6L7OKvKdLFgXFO4oO/kkMhiY8WvXgNMlzKw9/rkuRbhMbaRcLXmTJCq5VjK8W0T1DOKMmuLuNiVbKyAJKBJkaSUx6BXU9AX0fkW02PjmSGZE8qaP3NAPRb7wKZMIAuN/PvqGsatdYLA4UhinnoPuH7zyxFDgWr0xvcGjfGuNAG/fMSMRd7z0Qq4Dw7S1jVeNZ9tv/MqTU3Yp4H/gtSkDqUqT+6E4LdV6HCMu/wVaTPbLHjJL7rLP8GNLQdzGeZmc3OKdQ8HtkCpmeNqbdOmGkgKNhRr4jjqGc+8RDTLn+ik8wNuR/1c0RiXYKr2UXR4C3A1dbXbN0ShJJiwjzaMkvn7ZB4K6mL0i74TTX7ru54Gz3/oKlAOvnbX0qsYvndn9Stl42wXnxH6eUVCqdHVPd1eJ5kxqwm4/v0zI/ZCzIJpEjbm87Tqr4Q/g2Gqq4cv8JBRIxI5PAuTvqQuQljmiv6vDKwgdKjB3q+/mZvH4dLS6A0jzh8RiEhhoqbmkD7rTl5/vlXkF09ePkLmD9oxQDt4cZyCVcNtPTgUI+u09RpeKG+PP2eTpZovx6Bn6Ydea5SYQKklgDzIVq4omKVlMbOb4hZQUb/4gUzbG+rOq/ss4L8qKcddXzpLNjtXBH+9yYeIJPOHfxvrbzLCHpbZjGP8T53vqBnGyd9dTFGxXWbMa/Cn5k/2kJfDlPkBHR1BhbQ0TlPxxxcXeJsPn7V8HPD8N2O2Lnr8MGdXZnr2LIBkb17+ZEWT/bknP6hF/PnH3wULmnbqc36E59ve8+wYXUv/N0anhLdIdH6h84g2g=

zZ4Wvr6+4ffv2P54r17TUfAZ++uknsWXLFumxK1asEJ9//nm58540HceOHStUKpVYs2aNmD59uvjhhx+EEEL
MmjWrwvNNCCGOHDki2rRpI2bMmCF9r8XFxQk3Nzdp7x8hyj+eREZGiqSkJOnv7du3iw0bNoiAgACxfv16kZ
GRId13//59sW7dOuHj4yNt3SvPZ/tJtUVFRf3jfjnHE23yNCo6nvxbVlFRkQgMDBRnzpyRdTwpnadZTvLy8qQ
9IOQcTx7P0oxXx44dE927dxcbN26UHlfRseRJeaXnW1RUlKzjSem8M2fOSLdv3bq1xGesouNJ6SzNtMzOzhY
//PCD+PLLL2UdTw4fPiz8/f2FEEJkZmaKoKAg8dNPP4mEhATZxpN/y1q4cGGJ9y+EfOsm/5a3adOmEo+Rezy
pbFXvKGoZRUdHw8zMDB9++CHatWsHd3d3hIWFSddqBYBYu3brBysoKf//9t3QiBs1Jqsryy23prD59+mDfvn
3SWehSUlJw8uRJ9O7dGwAQGhoqnUW6rFlPyuvduzcOHDi48eePQ6FQlDiLXLdu3aRdFGrXuSWuUWozd3dHeHh4Th
//jzS0tKwYMECeHp6olu3blAoFGjevDkUCoV0EoGK1tanTx/8/PPPPiIuLw9KlS2FhYSEdL9O0aVOpNrny+vbti9
27dyM+Ph6NGjUqca3KXr164YUXgAg37Q8cOAAjh07BqD4GJratWtLJ0TQ0JwJVY7adu3ahdOnT+PatWvvS
Mg8AAwcOhJWVFVQqlWyfgV9++QU5OTk4ePAghbt27d6NZs2ZwcXGRfkEty3w7efKdJz8rcO7cO7cO
QDFJ/8aNmwYHjx4gJMnT8LT0xPe3t7S8+7fv1/iMira1vZ8hlSHS47t27YpatWohIC9+/L1Pe0
7I+/PBDFBQUSFlRUVGoXr06OnToogG3btmH48OHSyfweP5asInmFhYUl8jSys7PRrVs3WvLz8/Hnj17YGFh
gfHjxyM1NRW//fYbbt68ibNnz5Yp61+l5w4YNQ2ZmJnbs2ibnz5Yp61l5w4YNQ2ZmJnbs2IG2bdviwIED0niZkZG
ncO7776L33//XXpeZmYmunTpUqY8bZaTn376uLsLDw2FiYiIldzwgADx8+hLu7u
2y1DR06FOnp6fjll18watQoTJo0CZ999hmA4hNNtm/fvlyl13b17V/o7KSkJU6dOhZ2dHS5cuIDMzEw0bdoUv
Xr1wqxZs6THlXU8uX79Ovr06YPVq1fD19cXoaGhKCwshLOzMz766CN07doVly9fLvFd4+DDggDZt2iA7OxsnT5
4EoP1n+1m1RUVFlbhixoULF2QZT7TT7TJy83NlU4oCZ/PHlaVkZGBkxNTfHgwQNMmDKctJRkYGrK2t4
rqikOHDlVoPHla1rlz5zCTk4OOo78W9owlz8orvZzZzYBOHjwjoPS48owlz8orvZzZzY2trKMp78W9W9758+eRkZEBoPgz1DgHDiaGZBMnTgRQ/A0ATyhuYWGBnj17YtOmTUIH8+fPYWdnh6pVq+Ls2bMICQmBsbExrl69iqdPn8LW1hbV
5GnTMj09HXXZ2dvjss88weJzcYTzXp9cnIysrOzAQDVq1dHQkICFoFGjVqVOHx5FlZ8fHX4p4cmvWriqyba

Pj5Qq9VISkpCZmZmiXMelDVP07w6OjpKzSsAfP3117Czs8OAAQOk4/LS0tK0Gk+KioqwfPlyJCcno0+fPtixYwf
q1asHHx8fFBQU4PTp01i7di1mzpwpnXX34sWL2LhxI7p164b33nuvzMf0lac2Dw8PODo6YuPGjfi///u/Co8n
z8obOHAgnJyckJiYiOTk5AqNJ0/LsrGxwbBhw1CnTh0AkGU8KUtt9+7d03o8KWuWra0tPDw84OTkhPT0dNy
/f1/rsaS8tX3wwQfSDzhyjCfa1pecnIykpKQKjSfazrekpCQkJydXeDzRjF+1atXChAkTABR/v3l7e2Pr1q3S+rEQA
tevX0dKSopW6yflzXr8bOllUZHadu7cicWLF2Px4sVajydVnUFvWdZc1sHBwQHHjx9H9+7d0aNHD+Tl5SExM
REhISEICQlB9+7dER0dLX0ArK2tS/wKYmdnJ52aXhdZFhYWaNmyJby9vTFixAid1qa5TMGgQYMwaNAgnWa
ZmpqicePGAIq/jExNTbXaKlnR2jw8PODh4aFVbXLkubm5af1jQ0WWk9q1awMA+vXrh379+uk8T61Wo0+fP
ujTp49epmPv3r2lwxCepfTKZkREhPRl3aVLF1hZWWH9+vVo06YN6tSpg0WLFuHLL7+EjY0NCgoKUKNGDVh
bW6Nx48bSMqqLPGtra+Tk5KBhw4Za1VXRrPz8fNSsWRODBg2S9nDQ5bR8+PAhbG1tUbt2bTg6Ouosy9b
WFllZWVKGtns3VLQ2Gxsb1KpVS+stdxXNs7Ozg7W1NapXry41SXJnaZZJzXKv68/A48uJtbW1dJiKrmp7fDrW
rVv3mVna5FlaWmLt2rXo2bOnNF/69euHjRs34syZM3BycoK9vT3s7e2fOS3T09Ph4+OD+vXro0ePHvj111/h
6uqKK1eu4Pbt23BycoKrqyvatm2L3377DVOnTgVQvBXZxcUFpqamZb7UTnlrO3XqFBo0aIDBgwfjk08+0Xne
6dOn4ejoCGdnZ2lrr66yjh8/Lv3oINd4UlVq02Q5ODjAwcHhmWNJRfP+/vtvaVrKNZ5oU1/dunW1+ozLkaXZ
C0AbT8vr2rUrrKyssGbNGsTGxqJ58+bYv38/3njjDZiZmeGbb75BcnIyfH190bRp02dupJMjy9/fX+uGWY68ad
Omad1/GAqDPmZ5+fLlmD17NgCgSZMmuHXrFpRKJWxsbFCjRg3Ur18fK1asQJ06dRAXF4e8vDz88ssvKCgo
QPXq1fWWVaNGDTRu3FjrRrkieYWFhahWrZpeawMerdxq8wWvz9r0nSfHtDS22sqalQJ06dRAXF4e8vDz88ssvKCgo
69exe3b9+GlZUVWrRogVdffVXaSn3w4EHMnTsXXl5eaN68OczNzaHtTjNy5Gl2ddJXlrbXn9bntNTndNR3bZy
WVas2OeebpaVlieZVo3Tzqq309HSkp6fD19cXvXv3RpMmTXDhwgU0bdpUusZwRo1UK9ePSiVShQUFEjniB
g3bhz69eun9WE3ctVma2ur17yqlsXaDDevqtbWrl07hIWFASjeLfny5cvn9w9PREeno6vL29n7mBTs4sbRtlufK0
/RHTkBjslmWlUonIyEhcvnwZp0+fRocwDgRubm5cHFxwfDhw7Fnzx4x4AxQekz5w5E0lJSfDz89P
q12+5smrVqmW0tWm2hlbF2vSdx9kydKsbH733XcAinf7rkydKsbH733XcAinf7On/+PFxdXbFr1y5MmDABNWvWRN26dRn26dREfHw8
nJycEBwfjxo0b6N+/P9q0aOaajfPNZmmLXpO4+16bc2TfOakJCAgoIcmJmZwdTUFO6P6En0a9vb2ePf
dd5GSkoI6derA0dERSUlJaNOmDX799VdERESge/fusLCwgFKpLHFJpbJm6bs6feaxNsOzTd95VbIVbm2+Ph4AM
V7Xubk5GDOnDnSJcuqWlZl5BkSg22W09LS0KFDB3Tt2hWrVq3CunXrsGDBAsTExCArKws9e/bE33//bE33//jby8PA
DAzJkzoVQqtd4iU1lZrM1w81ibPFllXdkEgPr160tnRa/KeazNMGvTdx5rq3p4aKtK8K8AsVn7h4wYIC099qvv/4Kd
3d39OjRA7m5uQgMDMSxY8fwxx9/SCc5LMtu15VZmz7zD5s5fdJ1a8zWc6kspKZ1VGnkHR9bW
pdKWgoEDs379fCCHE+++/L13fMS0tTbqmqmfpO8+Ya9N3Hmsz3NqiNKXKCO0qmfpO8+Ya9N3HmuTJ0utVovMzEzpby8vL
xESEiIKCwvFrl27hJubm5g3b55wc3MTYWFhcrSdx5rM8za9J3H2iq3NjnGZyYEuHXrlujevbvIzjevbvIzs4WQghgx48Ughx48Y
NERERIcLCwkRiYmKFX1/ftekzj7UZOE2gwhqzLyDInBNsuPO3TokOjdu7dIS0sTSqVeHl5iU8++++USMG
jVKpKamGmyWvvWvvOMuTZ957E2eeeh6ZbMy81gb86palr7zjLk2Ik2IYQ4duyYCAgI44duOFSkpKbLn
CGHc05K1Ma+qZek7z5hrMwQGuxu2hlqtRs+ePRk5Ok5ODhwcHQQwUcHAw2S995xlybvvNYm
3wSExPh5uaG1NRUTJ4J48Gd+WrV4e3ZJZ++uinmsjXlVLUvecgwDExMRgy5YutHuHjxIgYPHwrfrJAcw7m
nJ2phX1bL0nWfMtRmCbREyu7WK0qzO8XVq1dF//79RUGsRWGdxwbcrk8+6detXk8+6detEixOs/
Sdx9qYV9Wy9J2nrYnrNWoUR2bMdsO9xb1wXvUsfecFecZc22sYoEeZ8WzsHhls995dXEwDBg0Q8
5xlzbfrO499ohY23yCA0Nxb1ZlePcNxOVHNwOUe5ODEQ2Vaw9ZlT3EVWHMU9L1sa8qpal7zsrjrs0QGEWz
TETGSZ8rm/rOY23Mq2pZ+s4z5tr0zZinJWtjXlXlXLL0neeMddmCNgsExERERERERZViUtlvgIiIiIiIiIiKiqYbNMRERERE
REVAqbZSIiIiIiIqJSDP46y0RERFRSz549cefOHelvc3Nz1K1bF7169cL48eNhZ2en1evExXh9u3b6NGJh47eKRE
RUdXFLctERERGaPTo0Th69CiOHj2KvXv3Yvr06Q0GgLC8OoUaOgVCq1eo3PP/8cFy9e1PE7JSiqq4ZZmIiMgI2
djY4IUXpXpD+btiwIVxc4XOxOxDh4YHQ0FAMHTr0ma/BC2C4YQEdHzjFuWiYiInEOtWrVC+/btFfT+tLtlghG6VpPPOxCEHhgT
Zt2uCVV17Bhx9iAsXLgAAPvroF/iroI9y6dQsrV65Ez549049AQBKpRJcBQUHo2rVr3r2Vrh+HDh+PcuXOsV
kmliJ6jjRv3hyhxsbG4ECEpkyZgZgDOEcpkyZgZgDGEjOnLkyHDJ0AEMdCDX1xcAsGLFCjg4dAmHkAdJkiq7OPTeDeNz8dlxhxYWAB
htDQULzxxhv4+OOPERkuEBMRKQO6N4ERERk6o6AOnMbgsG3GcuINetFdHPSYt7MvvCnhMPAiYT0Vm8BxsVDQYCHh9aB

AjRo1YGpqChsbGzg4OODmzZvYt28fgoKC8Nprr6Fx48aYMGEC2rdvjw0bNlRyVURERPLjMctERETPkdzcXNj
b26Nly5awt7fHDz/8gLi4ONy8eRNXrlyBWq1+4vMuX74MAPjggw9K3K5UKrU+YRgREZEhYbNMRET0HImOj
karVq3w999/Y/To0XBzc0O7du3g4eGBGzduwN/f/4nPMzc3BwBs374dVlZWJe6zsLDQ+fsmIiLSNzbLREREz4
mYmBhERUVh/vz52Lp1K7p06YJly5ZJ9x87dgxA8VmwFQoFFAqFdF+zZZs0AAPfv30fnzp2l2+fMmYOmTZti+P
Dh+imCiIhIT3jMMhERkRHKy8tDamoqUlNTkZiYiF27dmH06NF4/fXX0b9/fzg4OODq1as4d+4+4cEhMTsXnzZg
QHBwOAtFu1ra0tbty4geTkZLi4uKB3797w9fVFREQEbt26haVLl2L79u1o2rRpZZZKRESkEwrBiygSEREZlZ49e
+LOnTvS37a2tnB0dETv3r3xySefwNraGunp6fDx8UFkZCRMTU3RokULDB06FFOnTsWWLVvvw2muvYffu3Zg
7dy5MTExw4sQJ5OfnY8mSJdi/fz+ys7PRtGjTjB8/Hm+//XYlVktERKQbbbJaJiIilIiIlSuFu2ERERERERERESlsFkmIiIiIi
IiKoXNMhEREREVEpbJaJiIiIiIlSmGzTERERERERFQKm2UiliIiIiKiUtgsExERERERZXCZpmIiIiIioFDbLRERER
ERERKX8P6m21eelTBlfAAAAAElFTkSuQmCC",

"text/plain": [

"<Figure size 1152x576 with 1 Axes>"

]

},

"metadata": {},

"output_type": "display_data"

}

],

"source": [

"import seaborn as sns\n",

"import matplotlib.dates as mdates\n",

"\n",

"# Set the style\n",

"sns.set_theme(style=\"darkgrid\")\n",

"\n",

"# Increase the size of the figure\n",

"plt.figure(figsize=(16, 8))\n",

"\n",

"# Create the plot\n",

```
"plt.plot(data_processed['DATE'], data_processed['SALES'], color='dodgerblue', linewidth=2.5)\n",
"\n",
"# Set the title and labels\n",
"plt.title('US Retail Sales Over Time', fontsize=20, fontweight='bold')\n",
"plt.xlabel('Date', fontsize=15)\n",
"plt.ylabel('Sales (in Millions)', fontsize=15)\n",
"\n",
"# Formatting the date on the x-axis\n",
"plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m')) \n",
"plt.gca().xaxis.set_major_locator(mdates.YearLocator())\n",
"\n",
"# Rotating date labels automatically\n",
"plt.gcf().autofmt_xdate()\n",
"\n",
"# Highlight the highest point in the plot\n",
"max_sale = data_processed['SALES'].max()\n",
"max_date = data_processed['DATE'][data_processed['SALES'].idxmax()]\n",
"plt.scatter(max_date, max_sale, color='red')\n",
"plt.text(max_date, max_sale, f'Max Sales \\n({max_date.strftime(\"%Y-%m\")}, ${max_sale}M)', \n",
"         fontsize=12, horizontalalignment='right')\n",
"\n",
"# Create a grid\n",
"plt.grid(True)\n",
"\n",
"# Show the plot\n",
```

```
   "plt.show()\n"
  ]
 },
 {
  "cell_type": "markdown",
  "id": "ac54bcab",
  "metadata": {},
  "source": [
   "## 2. Split this data into a training and test set. Use the last year of data (July 2020 – June 2021) of data as your test set and the rest as your training set."
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 6,
  "id": "70fd1a07",
  "metadata": {},
  "outputs": [],
  "source": [
   "# Train-test split\n",
   "train = data_processed[data_processed['DATE'] < '2020-07-01']\n",
   "test = data_processed[data_processed['DATE'] >= '2020-07-01']\n"
  ]
 },
 {
  "cell_type": "code",
```

```
"execution_count": 7,
"id": "d5342d16",
"metadata": {},
"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>YEAR</th>\n",
"     <th>MONTH</th>\n",
"     <th>SALES</th>\n",
"     <th>DATE</th>\n",
"   </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"   <tr>\n",
"     <th>0</th>\n",
"     <td>1992</td>\n",
"     <td>JAN</td>\n",
"     <td>146925.0</td>\n",
"     <td>1992-01-01</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>30</th>\n",
"     <td>1992</td>\n",
"     <td>FEB</td>\n",
"     <td>147223.0</td>\n",
"     <td>1992-02-01</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>60</th>\n",
"     <td>1992</td>\n",
"     <td>MAR</td>\n",
```

```
"     <td>146805.0</td>\n",
"     <td>1992-03-01</td>\n",
"    </tr>\n",
"    <tr>\n",
"     <th>90</th>\n",
"     <td>1992</td>\n",
"     <td>APR</td>\n",
"     <td>148032.0</td>\n",
"     <td>1992-04-01</td>\n",
"    </tr>\n",
"    <tr>\n",
"     <th>120</th>\n",
"     <td>1992</td>\n",
"     <td>MAY</td>\n",
"     <td>149010.0</td>\n",
"     <td>1992-05-01</td>\n",
"    </tr>\n",
"  </tbody>\n",
" "</table>\n",
" "</div>"
],
"text/plain": [
"     YEAR MONTH     SALES       DATE\n",
" "0    1992   JAN  146925.0 1992-01-01\n",
" "30   1992   FEB  147223.0 1992-02-01\n",
```

```
    "60   1992   MAR  146805.0 1992-03-01\n",
    "90   1992   APR  148032.0 1992-04-01\n",
    "120  1992   MAY  149010.0 1992-05-01"
   ]
  },
  "execution_count": 7,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "# show the train data\n",
 "train.head()"
]
},
{
 "cell_type": "code",
 "execution_count": 8,
 "id": "977b2bdc",
 "metadata": {},
 "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>YEAR</th>\n",
"      <th>MONTH</th>\n",
"      <th>SALES</th>\n",
"      <th>DATE</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
```

```
"    <th>208</th>\n",
"    <td>2020</td>\n",
"    <td>JUL</td>\n",
"    <td>481627.0</td>\n",
"    <td>2020-07-01</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>238</th>\n",
"    <td>2020</td>\n",
"    <td>AUG</td>\n",
"    <td>483716.0</td>\n",
"    <td>2020-08-01</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>268</th>\n",
"    <td>2020</td>\n",
"    <td>SEP</td>\n",
"    <td>493327.0</td>\n",
"    <td>2020-09-01</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>298</th>\n",
"    <td>2020</td>\n",
"    <td>OCT</td>\n",
"    <td>493991.0</td>\n",
```

```
 "     <td>2020-10-01</td>\n",
 "    </tr>\n",
 "    <tr>\n",
 "     <th>328</th>\n",
 "     <td>2020</td>\n",
 "     <td>NOV</td>\n",
 "     <td>488652.0</td>\n",
 "     <td>2020-11-01</td>\n",
 "    </tr>\n",
 "  </tbody>\n",
 "</table>\n",
 "</div>"
],
"text/plain": [
 "    YEAR MONTH     SALES      DATE\n",
 "208  2020   JUL  481627.0 2020-07-01\n",
 "238  2020   AUG  483716.0 2020-08-01\n",
 "268  2020   SEP  493327.0 2020-09-01\n",
 "298  2020   OCT  493991.0 2020-10-01\n",
 "328  2020   NOV  488652.0 2020-11-01"
]
},
"execution_count": 8,
"metadata": {},
"output_type": "execute_result"
```

```
 }
],
 "source": [
  "# show the test data\n",
  "test.head()"
 ]
},
{
 "cell_type": "code",
 "execution_count": 9,
 "id": "71642016",
 "metadata": {},
 "outputs": [],
 "source": [
  "# Check for NaN or Inf in the test dataset\n",
  "test['SALES'].fillna(test['SALES'].mean(), inplace=True)"
 ]
},
{
 "cell_type": "markdown",
 "id": "66b830d1",
 "metadata": {},
 "source": [
  "## 3.Use the training set to build a predictive model for the monthly retail sales."
 ]
```

```
},
{
 "cell_type": "code",
 "execution_count": 12,
 "id": "939445b7",
 "metadata": {},
 "outputs": [],
 "source": [
  "# i am using a SARIMAX model here as it's suitable for seasonal time series data.\n",
  "import warnings\n",
  "warnings.filterwarnings(\"ignore\")\n",
  "from statsmodels.tsa.statespace.sarimax import SARIMAX\n",
  "# Define the model\n",
  "model = SARIMAX(train['SALES'], order=(1, 1, 1), seasonal_order=(1, 1, 1, 12))\n",
  "# Fit the model\n",
  "model_fit = model.fit(disp=False)"
 ]
},
{
 "cell_type": "markdown",
 "id": "75ea6efc",
 "metadata": {},
 "source": [
  "## 4. Use the model to predict the monthly retail sales on the last year of data."
 ]
```

```
},
{
 "cell_type": "code",
 "execution_count": 13,
 "id": "5cb53096",
 "metadata": {},
 "outputs": [],
 "source": [
  "# Make predictions\n",
  "predictions = model_fit.predict(start=len(train), end=len(train) + len(test) - 1)\n"
 ]
},
{
 "cell_type": "markdown",
 "id": "a88eaaa4",
 "metadata": {},
 "source": [
  "## 5. Report the RMSE of the model predictions on the test set."
 ]
},
{
 "cell_type": "code",
 "execution_count": 14,
 "id": "559e52bd",
 "metadata": {},
```

```
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "Test RMSE: 50429.681\n"
   ]
  }
 ],
 "source": [
  "from sklearn.metrics import mean_squared_error\n",
  "from math import sqrt\n",
  "\n",
  "# Calculate RMSE\n",
  "rmse = sqrt(mean_squared_error(test['SALES'], predictions))\n",
  "print('Test RMSE: %.3f' % rmse)\n"
 ]
 }
],
"metadata": {
 "kernelspec": {
  "display_name": "Python 3 (ipykernel)",
  "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": "3.9.13"
 },
 "vscode": {
  "interpreter": {
   "hash": "aee8b7b246df8f9039afb4144a1f6fd8d2ca17a180786b69acc140d282b71a49"
  }
 }
},
"nbformat": 4,
"nbformat_minor": 5
}
```