```
{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "colab": {
      "name": "Anjana. V",
      "provenance": [],
      "collapsed_sections": [],
      "toc_visible": true
    },
    "kernelspec": {
      "display_name": "Python 3",
      "name": "python3"
    }
  },
  "cells": [
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "5fCEDCU_qrC0"
      },
      "source": [
        "<p><img alt=\"Colaboratory logo\" height=\"45px\" src=\"/img/colab_favicon.ico\" align=\"left\" hspace=\"10px\" vspace=\"0px\"></p>\n",
        "\n",
        "<h1>What is Colaboratory?</h1>\n",
        "\n",
        "Colaboratory, or \"Colab\" for short, allows you to write and execute Python in your browser, with \n",
        "- Zero configuration required\n",
        "- Free access to GPUs\n",
        "- Easy sharing\n",
        "\n",
        "Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](https://www.youtube.com/watch?v=inN8seMm7UI) to learn more, or just get started below!"
      ]
    },
    {
      "cell_type": "code",
      "metadata": {
        "id": "Q94_4iI8yEfG",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "b5c77c81-0fd2-4f0d-e399-cca98515b3ac"
      },
      "source": [
        "from keras.datasets import mnist \n",
        "data = mnist.load_data() \n",
        "((X_train, y_train), (X_test, y_test)) = data \n",
        "X_train = X_train.reshape ((X_train.shape [0], 28*28)).astype('float32') \n",
        "X_test = X_test.reshape((X_test.shape [0], 28*28)).astype('float32')\n",
        "\n",
```

```
            "\n",
            "X_train = X_train / 255 \n",
            "X_test = X_test / 255 \n",
            "from keras.utils import np_utils \n",
            "print(y_test.shape) \n",
            "y_train = np_utils.to_categorical(y_train) \n",
            "y_test = np_utils.to_categorical(y_test)\n",
            "\n",
            "num_classes = y_test.shape[1] \n",
            "print(y_test.shape)\n",
            "from keras.models import Sequential \n",
            "from keras. layers import Dense \n",
            "model = Sequential () \n",
            "model.add(Dense(32, input_dim = 28*28, activation='relu')) \n",
            "model.add (Dense(64, activation='relu')) \n",
            "model.add (Dense(10, activation='softmax'))\n",
            "\n",
            "model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
\n",
            "model.summary()\n",
            "model.fit(X_train, y_train, epochs=10, batch_size=100)\n",
            "scores = model.evaluate(X_test, y_test) \n",
            "print(scores)"
          ],
          "execution_count": null,
          "outputs": [
            {
              "output_type": "stream",
              "text": [
                "(10000,)\n",
                "(10000, 10)\n",
                "Model: \"sequential_3\"\n",
                "_________________________________________________________________\n",
                "Layer (type)                 Output Shape              Param #     \n",

"=================================================================\n",
                "dense_9 (Dense)              (None, 32)                25120       \n",
                "_________________________________________________________________\n",
                "dense_10 (Dense)             (None, 64)                2112        \n",
                "_________________________________________________________________\n",
                "dense_11 (Dense)             (None, 10)                650         \n",

"=================================================================\n",
                "Total params: 27,882\n",
                "Trainable params: 27,882\n",
                "Non-trainable params: 0\n",
                "_________________________________________________________________\n",
                "Epoch 1/10\n",
                "600/600 [==============================] - 2s 3ms/step - loss: 0.7849 -
accuracy: 0.7732\n",
                "Epoch 2/10\n",
                "600/600 [==============================] - 2s 3ms/step - loss: 0.2205 -
accuracy: 0.9374\n",
                "Epoch 3/10\n",
                "600/600 [==============================] - 2s 3ms/step - loss: 0.1620 -
```

```
accuracy: 0.9512\n",
                "Epoch 4/10\n",
                "600/600 [==============================] - 2s 4ms/step - loss: 0.1409 -
accuracy: 0.9575\n",
                "Epoch 5/10\n",
                "600/600 [==============================] - 2s 4ms/step - loss: 0.1145 -
accuracy: 0.9654\n",
                "Epoch 6/10\n",
                "600/600 [==============================] - 2s 3ms/step - loss: 0.0980 -
accuracy: 0.9702\n",
                "Epoch 7/10\n",
                "600/600 [==============================] - 2s 3ms/step - loss: 0.0893 -
accuracy: 0.9737\n",
                "Epoch 8/10\n",
                "600/600 [==============================] - 2s 3ms/step - loss: 0.0784 -
accuracy: 0.9768\n",
                "Epoch 9/10\n",
                "539/600 [=========================>....] - ETA: 0s - loss: 0.0685 - accuracy:
0.9795"
              ],
              "name": "stdout"
            }
          ]
        },
        {
          "cell_type": "markdown",
          "metadata": {
            "id": "GJBs_flRovLc"
          },
          "source": [
            "## **Getting started**\n",
            "\n",
            "The document you are reading is not a static web page, but an interactive environment
called a **Colab notebook** that lets you write and execute code.\n",
            "\n",
            "For example, here is a **code cell** with a short Python script that computes a value,
stores it in a variable, and prints the result:"
          ]
        },
        {
          "cell_type": "code",
          "metadata": {
            "colab": {
              "base_uri": "https://localhost:8080/",
              "height": 34
            },
            "id": "gJr_9dXGpJ05",
            "outputId": "9f556d03-ec67-4950-a485-cfdba9ddd14d"
          },
          "source": [
            "seconds_in_a_day = 24 * 60 * 60\n",
            "seconds_in_a_day"
          ],
          "execution_count": null,
          "outputs": [
```

```
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "86400"
            ]
          },
          "metadata": {
            "tags": []
          },
          "execution_count": 0
        }
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "2fhs6GZ4qFMx"
      },
      "source": [
        "To execute the code in the above cell, select it with a click and then either press the play button to the left of the code, or use the keyboard shortcut \"Command/Ctrl+Enter\". To edit the code, just click the cell and start editing.\n",
        "\n",
        "Variables that you define in one cell can later be used in other cells:"
      ]
    },
    {
      "cell_type": "code",
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 34
        },
        "id": "-gE-Ez1qtyIA",
        "outputId": "94cb2224-0edf-457b-90b5-0ac3488d8a97"
      },
      "source": [
        "seconds_in_a_week = 7 * seconds_in_a_day\n",
        "seconds_in_a_week"
      ],
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "604800"
            ]
          },
          "metadata": {
            "tags": []
          },
          "execution_count": 0
        }
```

```
    ]
  },
  {
    "cell_type": "markdown",
    "metadata": {
      "id": "lSrWNr3MuFUS"
    },
    "source": [
      "Colab notebooks allow you to combine **executable code** and **rich text** in a single document, along with **images**, **HTML**, **LaTeX** and more. When you create your own Colab notebooks, they are stored in your Google Drive account. You can easily share your Colab notebooks with co-workers or friends, allowing them to comment on your notebooks or even edit them. To learn more, see [Overview of Colab](/notebooks/basic_features_overview.ipynb). To create a new Colab notebook you can use the File menu above, or use the following link: [create a new Colab notebook](http://colab.research.google.com#create=true).\n",
      "\n",
      "Colab notebooks are Jupyter notebooks that are hosted by Colab. To learn more about the Jupyter project, see [jupyter.org](https://www.jupyter.org)."
    ]
  },
  {
    "cell_type": "markdown",
    "metadata": {
      "id": "UdRyKR44dcNI"
    },
    "source": [
      "## Data science\n",
      "\n",
      "With Colab you can harness the full power of popular Python libraries to analyze and visualize data. The code cell below uses **numpy** to generate some random data, and uses **matplotlib** to visualize it. To edit the code, just click the cell and start editing."
    ]
  },
  {
    "cell_type": "code",
    "metadata": {
      "colab": {
        "base_uri": "https://localhost:8080/",
        "height": 281
      },
      "id": "C4HZx7Gndbrh",
      "outputId": "46abc637-6abd-41b2-9bba-80a7ae992e06"
    },
    "source": [
      "import numpy as np\n",
      "from matplotlib import pyplot as plt\n",
      "\n",
      "ys = 200 + np.random.randn(100)\n",
      "x = [x for x in range(len(ys))]\n",
      "\n",
      "plt.plot(x, ys, '-')\n",
      "plt.fill_between(x, ys, 195, where=(ys > 195), facecolor='g', alpha=0.6)\n",
      "\n",
      "plt.title(\"Sample Visualization\")\n",
      "plt.show()"
```

],
"execution_count": null,
"outputs": [
{
"output_type": "display_data",
"data": {
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAAXoAAAEICAYAAAABRSj9aAAAABHNCSVQICAgIfAhkiAAAAAlw
SFlz\nAAALEgAACxIB0t1+/AAAADl0RVh0U29mdHdhcmUAbWF0cGxvdGxpYiB2ZXJzaW9uIDMu
MC4zLCBo\ndHRwOi8vbWF0cGxvdGxpYi5vcmcvnQurowAAIABJREFUeJzsvXe4JOdd5/v9VejuEy
dogkbBki1L\nloUlW7IALWYXgw2XLFgvcAnGpDULvg/2xXjx8rCENXgNlzXBrGG9zlg4YBks27JXsiy
hHGYUZjQa\nTdDkmZP7dK5c7/3jrbdSV3VX9+kzJ8z7eZ55pk+f6urqPlW/+r3fXyLGGCQSiUSyeVH
W+gAkEolE\nsrpIQy+RSCSbHGnoJRKJZJMjDb1ElpFscqShl0gkkk2ONPQSiUSyyZGGXrLhIKI/IqLP
rNK+/56I\n/utq7Dv2Hg8Q0a8Fj3+eiO5Zhff4PSL66Kj3K9mYSEMvKQwRfTcRPUpEdSKqEtEjRPTta
31cRSGi\nbxDRf8t4/nYimiUijTH2nxjsYTDTXJ9kMx9gMr2QcRvZGIzqb2+37G2K+t7OgkmwVp6
CWFIKJp\nAF8F8CEA2wFcDuCPAVhreVwZFV5aXwBEsmqIw29pCjXAQBj7LO
MMY8xZjDG\n7mGM7QcAIrqGiL5FREtEtEhEdxDRVvFiIljpJRO8hov1E1CaijxHRbiL6OhE1ieibRLQt2
PZqImJE\n9HYiOk9EM0T0O3kHRkS3BSuNghE9R0RvzNn0XwBcAuDfxl67DcCdCPAvh08PMniehPgs
c7iOirwX6r\nRPQQESnB7xgRvTK2n/jrtgWvTWyCi5eDxFTnH/ktE9HDw+D8TUSv2zyGiTwa/+2UiOh
R8V8eJ6NeD\n5ycAfB3AZbHXXZaWt4jox4noYPBZHiCiV6f+Nr8T/G3AdD2/DHZBkKKCvd2pDRBGf+18
BQCDDfI6IzhFRA8BnAOxI7WIu9tjI+Hkytf2ZZ\n2ONTwfuluQrATwWSRI2IauA3pD05n+IrgHwWSRJZ
HwHQD+Mecj/38AjgG4J5BL3puzXQIiGiei\n/0VEp4Lv4v4kEAW4lILfJ6AB8DcJgx9mexff4QET0eSEg
18BtZ+vvN4zLw7w8AwBjzwb/by2PbzMYe\nd9D9t5BsYKShlwwFY+xFAJ8EN/gA8H5wbxw/9Gxtg0u
KedDnoOypWxxy8DcD5jmzMA/oExtjX2b4Ix\n9oEe+/00uCf/CwD+CwD+D2NsLmsjsxjxliTMfZuxtgrAPw4g
N8mojcFv+4AGI9tf1fb8L\nnu/7fQQ3k+sA/GrsuA/GrsuTKAO8E98d2Msa3g8ovYX79V
xnnwG6LYH4F/t+f6HY9kcyANvaQQRHQ9Eb1b\nBBWJ6EpwKeMxpAC0A9UA3f88I3va/Bt7xt
4Hr0p/P2OYzAH6MiP4vIlKJqEI83zTAz+BnwaQBv\nBvAfkS/bgIh+IIheGRjGOgAPgB8+l+kAPxe85w
8iKc1Mga9QakS0HcxAfFvmwRPSTY+0tBLitIE8J0AniCiNriBfRbA7c\naAA81fIWcIP4NQBfGsF7/iu4
4dHIfgL9gjHUVFjH\nGzgC4HTxQugDu4b8HPc5txhJcCM3AeCuHu9/\nLYBvgt/AHgHgPwY/cHv3gkeUK4B+HlwTV3w
V+Aa+CL49/SN3h8zG5GfAg6Z/OfJ4RXbVOXaF/hDLUg5A/\nFzfmewF84y4OwP1ZvVhvv7Jba\bTIKGK/Ul+u7V/hASiUQiyaev\ndENEewDsYYw9TURTAPYB+Anwgooo
qY+wDQabANsbY7xLRjwN4J4AfA/DvAPwlY+w7\ng+jzUwD2ggfL
9gYCBfcbgPCrRlq5tSrOgZ6TzMxDEL1jQPPmDSQuwJmBw8eRDiz5JBOJ2PCuwZtfY+hwY3jiQ0
WYzuRGEeq\n/6KUjRWf/8QwGRUcpvG/qDL5YLL+EHjNh6Xdd8h7E/vkd8gFnO1BVGLvoDB0zqh9
MsEIa/zmA\nuVoBtYsdIoK/NzXBy7GjCUSbxJF24DgSTeDCUcyUHgTtt6GYyTLddudHOkrt7mybP8VK7TKEw0gaHYfB2Gg4Nb8C6zEg3JHcoCtiqhN0uH0jEfR0W8w/P0hhjhF8UNMfDYEySrVDMOJUxZzeFr74OsYQ4j/HmBQkVYkCXGMNq9mE\nhLg40j1WfHCFKZk5jU5zIuc6aVHgr0LLbGjmDuRZ1ZSVxQgH9w5Mm9Av7tPADfD3LKnHzPH4Bep0z\n2zoW0d6S4z5SynKgOUz3cRzgoEkxEtnQhzx4ZQSgGH5Ofg4BbIfx7DPPbzZjeoeTHPfYF\nR58tNqodXRVFRGXoH93o\nxzvIjw4AV8O3S6Q7DDX40LtC1AyFydR0GTSHSgeIz3L7mHsKC2jsqHhyE1n\nqYgr5b02eBMz1\nCv8nPh3c64rgoVmpvyRgUJS\naGhN4p8gvIrl7vC1y1f5N4hbPmgE9qMhy3fgXvt5W2v4cSxnmvz\nZFJbfm8xXlgE7Pq9zIg42x2r\nbSp5JBjjx3MrXHyB3V4o1yNsAgOq0BKtZnS\n/y0eFbqG8m5TLjqBhPxklUG2CqIbfdB8IJcMWakeZV3ExhzD7Z9A13NZzn4nqGlXmpdWQ0L8i\n8bSD5p5MbeHSQQD4z7XJYafIWtCzb6kxJGDqGpQ33P4WeT4gnn0b60Wj9PK1HXtPVjgIH0sKx\nmxE7W4LgsKNLZzuf3ImsUkxPfiizm69nK2Bkk4Ewk+9ibdLRWFDQ06eS/OYJtgJbA3vYwD2Yq2anGFK\nI5eQqSaHNLZb/ud5d8iLpFQCpbHxzNjLbvNoPvwHcvt+RBIQDJhVi5lmWJ6OZ8A1xKpSxNMe2c/GJj4FpJ3v7Y6TlTQnmmr2gxb9vg\nHjwy55/7vg5Jz6sJvuUQLbX64cbH1Wy1Pkbf8ZE28u1T53nNEv4mSmAUvXJN3vJ4Zfhvd5Q5wsXHB4ADsuF2EcOK5+I\nLd7sSgOeh2q6P6mPhQNYhXyA4Njs4ySnXuYO7UDVH8RsqR9w3gx3XH\nI0LbpHyZtUmUo9lOrGBeXN+JRf2Aeog2jUz6dJt\ngpfTt6uHpHJDvhpdiw+C0hqp\nnY7Yss1YzpO4mcG9DNhcwKCnlR4hajhD\nWiW3WQZoIjPMj9bMZ4WNBxzL3KBeAhDn9+ZfIHIwYTD5rFpEjBMjZvv+AZXdGkrUUQXVtRjHFd0J5uzWZMjtjCQcRpYl7k7fnTlRmm+y\nCwZD9HJlI/K5OI6NySJxfg5GrC7vq5FLuSJDN5nG\nyrvKNqYmJPgHPJqx\n0BEbV9KsRxp5V7mZFmMaNK7HXeFXPm3MPQ2ecAkaeWfyIg0vKwTGCYGQeqKTzcXDXbLiRFXhRQP3pMsSSLyB9g48gdbrmXsqxy\nREkkepxVBMn1UEjJNOZZw==...

B4HsYYzcCeB+Aj4z+sNcWURWrqvx/GZCVXAgOnq/j0ZcWh3pttc0NvejR\nJBmeA2fruOmP78Hpp
c5aH0ph+hp6xtgMY+zp4HETwCEAlwO4HcCngs0+BeAngm0eZYwJq/c4gCtG\nfdBrjSE8eoX/Pz1
Zx3zTCS8mScRy28acrBweCX9z31H8wZefH+q1S8G5aTgbyxNdj5yuduD5DOfr\nxlofSmEG0uiJ6G
oANwN4AsBuxpiYvDELYHfGS34VwNdz9vV2ItpLRHsXFhYGOYw1Ryx/NZVfNKUS\nN2TLHWno07
zvay/gNz6zb60PY1PQMNywPfagVNsWAOnRjwLD4d+h6Wyc77KwoSeiSQB3AngXYywR\n0WGM
MQAstf33ghv6383aH2PsI4yxWxljt+7cuXPgA19L2qFGnzT4G023uxAstWzMydTTkdAw7aEN\ndej
RS0O/YsSKftMZeiLSwY38HYyxLwVPzxHRnuD3ewDMx7a/CcBHAdzOGFsa7SGvPeJiEdKNMPQy\
nda0b0/HC4LVkZTRNB6bjD/XaxRa/2UqPvhvH8wdy0sR3OOzfYi0oknVDAD4G4BBj7IOxX90F4G3
B\n47cB+HKw/csAfAnAWxljR0Z7uOuDdDBWDT16eRGlMRwXpr1xLoj1TMty4XgMrjf49ymDsfl86L
6j\n+MkPP1J4eyHdGBvIo9cKbPMGAG8FcICIng2e+z0AHwDwBSL6VQCnAPx08Ls/AHAJgA/zew
Rcxtit\nIz3qNaZjRemVQNyjH04/3QxYrgeFCLqa9B1Mx4Ph+PB9BkWhNTq6zUErcCQ6jodptXh4zf
V8NIzA\nOMnVVRcnlzoDZdCIFf1GksH6GnrG2MMA8q7QN2Vs/2sAfm2Fx7WuEV6RltLoWxexR/8
rn3gKr9w1\niT++/TWJ54WOaboexktF/ApJFrbrw3Z5GMywPUxX9MKvXe5EDoj06Ltpmg5M14fn
M6gFnBEjdk5v\nFGRl7BB0UumVqgzG4sRSC2eWu9PNTJfLDNLArIz4uTXodxlP++1sILnhQlE3+I2
waCwp1Og30Dkt\nDf0QtG0PCvlQFO5hqYoHgF3Uhr5puplLWTsw9BtpmbseaSUM/WDn2VKQWgl
EsqMkomEOFr8Q57Jw\nYjYC0tAPgWF70LToj0wE6Kp/0WbdMMbQtrzMYhyRmdCW2vCKaJor9+
h1bfj0zM1MI/Doi16/YTB2\nA32X0tAPQdtyQ31eoGnumnj0z5+r41svzl3w941juT581p2FwBgLPfq
VGpizyx28/n334vhCa0X7\n2ai0RiDdVModecPNoGkGQe6CMbbOZs2jlyTp2F6ozws01V2T9MoPP
3AMf3jXwZ7bPPbSEn78bx9e\ntRNTeJvp/bs+gx+U0a3U+zmx2MZS28ax+YvV0EcB1UEzZ0RDs7
GyIWsaUrieD2PAVaeQvzZSeqU0\n9EPQsV0oSjKVUlGcnks/y/Xwu1/cP/K+L8ttB60+ZfFPn17G/rP
1Ves5I1YyaUMf/3mlHr24UVys\n8thKpZuS7kLTHGnoU8S/16Ir8o4jPfqLgrbtQkl59Kra29AfnWvh8
3vPDN19MI/ljtX3wl8Olu7x\nNLtR0so19FEcY6UGRnhP69XQO56Pzz15Gp7P+m88BCuVbkq6DVX
xNpSufCFIGPqC382mrIyVdNOy\n3DB3XqCqLlpmfIMzcUKNOtd+uWPDcllPAyMM/Go1XRNGyEpII
Vju6Dx68fr4hbmeeOTYIt77pQN4\n/PjqdPxoxT73oMZ6qW1B10yoqhfKFBJOI7YaLpqRJJyOjbQ6k
oZ+CDqWG7Y/EGiq29PbFF0HWyM2\nVEVygGuBga+tlqEPPpPlMPD+dpykR7+5pZuFJk9hPJdRSz
AK+Ofm3+2g3+Viy4SuWVAVF3Yfp+Bi\nI27oi55boqWHlG42OR3bzQ7G9rgAxUk0yjYJpuPBdKJqy
TyqgYGvtldHuhFBLAbAjvVhiV8IKy29\nD6WbderRi8yW1epR3jRdlDQPquKHGnFRlto2SroVOicbKYi
42jSMwSQxxljowAz6d1hLpKEfgo7t\ndXn0ap+sG2HoR5mZI/J/gd76ouhFvloefVxOiTcwi0s5K5du
1neHUNEGeKa2OgHvpulC01xoqj+Q\ndOP7DPWOG2r0wMaSHFabeH//Ilk3luuH/dg3UrxDGvoBYY
zBcPwujV5TeWdBJ6ezoDCGo9SYa3FD\n38MACgO/Whp9/L3j/T+sEWbdrHeNXqQwrpZH37IcaKr
b16FIUzcc+AzQdTts1bGRDNRq0wjOJ6Ji\nrYrj53E6JrWekYZ+QGzPh+dH/W0E/YaPRB79CA19LIs
mbznu+ywsCFleJekm7mXH5RozEYxd2ec2\nw6yb9dkhVLQZOLvcXpX9t0wXihJkzgwgGYiVRkmz
Qo9ettOOEKvisZJdqGBKnMeaZkuNfjMjTgZx\n0QiE4c/zOIW23ByhoYpLMXk3klbphEVLq511AyQ
DsNYIg7Hi9Y0hR+mtNostbuhn61YiID0qmqYD\nTXWgKM5A36WIHZTiHn3sRiHqO87Viq9E/vwbL
+LOfWcLb7+eaZoudNWDpjmFpBth3Eua3ZV8sJ6R\nhn5ARPe/XI8+52QJs25G6dEb/dvPxnPnq7H
mVr1YbFl4/Z/ci8deKpYqGA+QZnn0mprd8GwQxOfr\nVxy2ViwFE5wMx08E+EZFI5BuFMUd0NDzv
3lJt8K2HfHXH5vn9R33HpwtvM9/2ncGXzsw03/DDUDD\ndKBr/HststIR311Jt8GwceQbaegHROTa
pnvd9GtVLAz8sMOdsyji0QsvvqSbYfZNPx4+uoillo1j\nBfvKxG9ucQlJePe61r+oqx/mOi+YqrYdVMp
8eMVq6PQt04Wq8bTe9gCrwqW4Rx9kisX/FiI991S1\n+OCNWsfBXGN1YhEXmqbpQNOcoOCx//c
qvjtd5zdQa4PUJUhDPyDhGMGu9EphiLINWlQwNUpD31+j\nFzeDibFWYvteCE++aAEJ/2x8CZvw6J
3l+1np526vQtbSqOjYLiyXYXqiDgCYWQVD37Y8HoxV3IHi\nHdVWXLrp7rooNOpTi8ViC6bjwfEYFjb
JwPeG4UJR7SDI3f97NWLSTfzn9U6RmbFXEtH9RPQCER0k\noncGz28nonuJ6Gjw/7bg+euJ6DEis
ojod1b7A1xowqEjGQVTQL5nLTz5URqqmuGEN5i8/YoA7PhY\nG7bLCkkoD7+0wPdZ0AtvWTx9D0
hp9MGyVtftFQdjxarBcvMzm9YKkXEzPVkDAJwfcYql54tMLweq\n6g20Olpq29A1D4rix9Ir44aef68
nloqt3oSzsNR24G+Cwqu6aUNXbWiKV8gZMUKPfpMZegAugHcz\nxm4AcBuAdxDRDQDeC+A+xti
1AO4LfgaAKoDfAvAXq3C8a04YjM1ogQDkSwvC0Hcsf2QBnHrHQaVs\nAWC5hlRlNxOVVuLnPM4
ud3Bu2QyOtahHzwtygGTbA+HR8z7oKzP08devtwEvQh6ZGm9CITZyj16c\nU5rmBlk3gwVjxU1YnK
Px71JIN2eXzUIVs2J7z0/GiDYqDcMO01aL3ECFoRce/UbJvOlr6BljM4yx\np4PHTQCHAFwO4HYA
nwo2+xSAnwi2mWeMPQVg458FGQjPMp1109ejt6Lq0VENf6h1bGiqCV3zc/dZ\n6zggMIyPcQ02Pl
YuiygAywb06LmhN1MavaLwmoOV9lgxHQ9EfB/rLZc+DHiWTIyVrZF79KGhVx2o\nqgvDLv5dij43Q
HTOGhkaveMxzBbobhqPC4m2DxsZXojGA92dAt9rJyZHApvLow8hoqsB3AzgCQC7\nGWMi9D4LY
PeA+3o7Ee0lor0LCwuDvHRg/uqbR7D/bG0k+4oGgw+WR881VrvnNoOy3LGhazY01evp\n0Zd0F7o
mqmN7338fO76Esu5gomIW9sJblodyYOiNREGJB03xB5YbsjDs6D3WW0B2MaaDl0ptnB8g\nVbEIl
qtJU/nAG8djcGPy1dG5Jt731RfwP+8/hi/sPYNDM43wd0stK/zbKwqDQn5ibmw95pUX0enj\nXvxG
N/SM8RoTHoztXfAoEK08dH1jefRa0Q2JaBLAnQDexRhrEEXT0hljjIgG0iMYYx8B8BEAuPXW\nW1d

N7HM8H3/1zaNoWy5uumJr4ncf+tZRXLZlDG95/RWF9ycMVlq6URQfCvmZwVjL5Rfn5LiJVqeE\npuVi1xCfJc1yx4JecqD2SA2rdRxugPT+HSwZY3jk2AK2Ti/AMKcKxRMYY+hYHi7ZFnj0brzXjR9o\nw1EzLVWhvF31xHB8TIybMO2xwob+4Pk6Lp2u4JLJ8lDvWZQoV91CuWTgXK14BksRhOwngrEA9yyn\nVe6nfWHvGXzs4RPh9goBd7/z3+L6S6ex1LZQGov+5ukWCnXDgap48HwVp6odfFefY4nfGBZaGzsg\n27E9XjWsuiAl6F9jedgynu//itXUppNuAICIdHAjfwdj7EvB03NEtCf4/R4A86tziCtDZBVkLfc/\n++QpfOaJUwPtT+jWaekGAHQtu4xaeGTlEr8wRuXR1w0XumYH+mK+R6+qFnStfxuEU0sdzDVsbN+y\nCFUtNqTCdPgYwVKpW7qxHN6Ea6XNtJygGjn06AtIN4wx/OxHHseHH3hpqPcchKWWxT+n4qFSNjDX\nsEYaqGzGNfqMzJnljoPxioU33/ZVvOHmb0FVXXzw3iNgjGG57YSpgACC9Mzo+2uYDibHW1AUHyeX\n+nv09c7m8eiFTRDSDdC/303HcYNzWrSTWF+JAXkUybohAB8DcIgx9sHYr+4C8Lbg8dsAfHn0h7dy\n6j0MfdN0cXi2MVBwtG3zDAZF6X6NlpOiJTzQSokv6UfRgdF2fXRsH7reu1qSa7R2ZOh7tEF4LOil\nvn3LIi8gKWDoI/3YhaL4XVk3iuKtuJmW+GziRtkscKNcbNlomC5m66vvdS61bZRLDoiAStmA47Ew\nQDsKIunGycyc4bEaC6rqY3K8hZftOYZ7Ds7hsZeW4PqRngxwQx+XbmodG5pmYbxi4PRS/5Vl3eAx\nH1XxN7yhb4QrJSdcKfVzwgzbg6pGGUybyaN/A4C3Avg+Ino2+PfDAD4A4PuJ6CiANwc/g4guJaKz\nAH4bwO8T0Vkiml6l4++LMPTp0nnGGNqWh47t4+wAPcQN24WuZt/F86ZMiZtMpcyNzig0ZvG5dM2G\n0iM1bLljQ9dtKAqDrnk9PfrHXlpEpWRjYqwV3LT6x9Pjhl5T/K48ekXxQm+p6PDlNEZo6It79KeD\nAqClgtXAK6HatsOA51hwMxpl5k38O87KnFkOjLXg6suOo6Q74SzhhKFXnMRqoGbY0FQHY+UmTiz2\nT7GsGTzmUylbG97QC0lMT3j0vc9Rw/agKd6KV6kXmr4aPWPsYQB5wuqbMrafBVBc9F5l8gx9O9Dn\nAODwbBNXbh8vtL92RotigaI4mR5BM5RuAo9+JIaeX7ziJM3zROodB3smo2BhXqtirs8vYuv0PIhQ\nuICkncgI8btaIBBFcsOwAVlh1IRHXyTf+Uxg6C9EYc9C0wwNfVgdWzNx04iugngwNitzhgflo+9E\n01xcfdlRHDl1AwDe0EygpAqu6oaDiUkHimLh1GIHjDHE429pah2Hn3Payg39YsvC3Qdm8Nbbrur5\nnnquFqCHQNBe+z9+/X0pxx+HXv7lJPfoNjWhD2jCSBi7ey/3wXLPw/rKGjghU1c1sWhZKNyP06EX2\nTC+N3nQ8WC4LMwQ0zUI1J+vmxGIbS20H27fwmbaqUixTRtzEVNWFoniZHr34vgbpuhjHSKS0sUIe\n/ZnQo1+dRm5xltpW6DWLv/EoPXohValq9k1TZF/FuXLPCVRK0Q1ewAP3QaovY2gaLnTNwXilDcPx\nwwyiPOoGN/Il3cRcc2Wf8SvPnccffPkgThWQjFaDuHQTVbb3PrdM24OiuIWkm/VU2LfpDX2eRh//\n+cXZQQy9B0XNNpZ8nGCWoefPVUrZhv6Bw/NYag3mHSUMfY5RFjKN8PZ01cptbCZ6nUyO8+W7WjCv\nOF3ME9foDYfr8yv16IX3qip8yllIRjV5IN/WOu6qj80TAU9QR8L+HP9IUy5bpQtc8vtIK+9VExrph\nuOHNXKCpHl5+xYsg8sNVBhBo9La48XpwfR6MHK/wQOypPgHZ5Y4NTbVRLpkr9uhFttKZ5bUy9EGq\nZJBeCfQ/Rzu2B0VxQMRAxJA3IPxMtYNv/9N78dGHjo/2oIdk0xt64bmnjavQ51TFxaGZ/Bz7pukk\nvLO25UKhbEOjqW6mtymeK+kWFEp6pB3bxa988il87qkzBT8RR+Qz67qTa5RF4FV4e7puYznH0M8F\nQUsRMBb52v28knZMPyZyEz3ouUfvZzbTGoQopdWDpnkDafQMq9eeWRxbfNVEBIyVTZyPBYGPzTcT\nee+D0rIc6Gqy9Yb4Ttq2B89HQroRXHnpSbzx2+9BuRT36CNDL6QLXXMwPiYMfW+jW+tY0DV+Y6sb\nHuwVdG8Uf5cz1bVpkNZMePTFsm7atgNF4TddTfFzNfr3fe0F1Dou/uwbL+LYfLH2EqvJpjf0wqM3\nnnaTREsu2rVPLOLHYSZTux3nfV1/AW/7u0fDntuV2FUsJ1Jy5sVF6nANNS6a3LbVs+CwpJRWhFnrr\ndmYRTXwbsXTXNTvRtjiOqIoUOnhR49xMZN0k2xFbXR79yqQbVXGhKtkB7zQnl1rhZ1jqI0esBOGV\nlmMpjPGiqX89soA3f/BBfOmZc0O/R8viYwSBqDBP/F2W29HfNg1RUrYB+DkqqpTjAf2xcgcE1tej\nrwerBxEYX0mwWzgiZ9fKozdcKOQnnJFCWTeKcDyyDf0jxxZxz8E5vGzPSyBy8N47n1vzvkCb3tDH\nDWhcrhGPt04vwfOBl+azT/CHji7gfM3EfBDUa9tR5kMaLShPT6drNs3ghCIfuuompAehIRdJZYwj\n0tzSRTRxljtJj76kO+jYfqYXNtcwUSk5YdpoVnZHFu2YfqwofkKHN92kRr9i6Ub1oKpO3xYItutj\nvmGHTcYGlcUGQQwciUsnlbKB87UO2paL9975HADgmdPLQ79HM5guBXS3MRDGupTh0WehKl7XazXN\nhaIwjFesnu2KfZ+hZXrQVAdlnV8PK5FvQo9+gKw3geP5uO3938SXnx3+BtowHZT0QBLr0xxQ0Ild\n/2oqJgUArufjD+96HuMVA9ddfQjXvfwA9p6q4Y4B63VGzaY39PWEoY8eixvAtukqAODwXANpZuoG\nZur8RH7hPP89D8ZmnwyqJn3SlXCY01lc0ijNCgqYeitUH8JE3vY7mT9PbE/zWj2/ubrZsolaIL\nLmtIRRYt0w3yqr2uE98SlbEZRT6DEEo3ileob/i5mgEGvloDVjcgG6+KFVRKBhaaNv7sGy9ipm6i\nUu6sqAVH0+T90gF0acm11M28H6rKpSbPZ5FHH7TmKPdJsWyaLhi4XCgK5FZi6MVq4HR18PGLyx0b\nsw0Lz5xeyffqQgu+VyL0bCUi6DiRR59OPgCAO544jWPzbVx39QGoio/Ldp7Bjq3zeP/XDw00xWvU\nbGhDbzoeHj++1HO8XD3HoxeAI2v+AAAgAElEQVSBmC1Ty1DIzwzI7j0ZeWEHQ0Pv9fToge54QHzp\nrSh2QmMWssKgHr1IcwNiQ09S+4ikm8DrC7zOrH43M3UDZT0etCuW+96yopuYkuqsaLlsJMHYqDU0\nnl836DW8R+vwWYehX0aOPD/YQVMoGfAZ8+rFTuPLSk7h0xzkcnm0OrWc3zEhD5gFZH51g5RTezPWi\nHn0UiI2km6CVdaXdU7qpxVJ6yyMw9CJedGYIQy8ctZUEvRuGA1WNt4fwcudJCEwnclwUxU0EY9uW\ni/9xz4u4ZMsCdm3nE7uIgBuueQ6m4+FzT54e+lhXyoY29AfO1fF/f+RxPN5j5F2tE03Wid8Q+HLY\nh6Z6mBxv4/BMlqGvQlM9jFdMvDDDK2gN2++aLiWIGpsIf980I3klnYK5GJzsgxrBWscJi2Q0cfHa\n3dKNpvKAKBB5fVkdLGcbZqjPA5FE0O8GxG9ikcdtBYbe9xkcj0FRPCjEKykHvZkJTCfav6Y6fefu\nCkM/PVkDYbRVqmmWWhkefZkbn7GyieuuegHTE3W4PnBkgDTeOK1gXqwg3q+mZgzu0QP85tkIpZvI\n0NcNL+EcxYmvHkRMYlhDzxhDzXBB8FFtDz5qUhzjSvoKNbq+194ePWMMph3p+em

Y1MmlNhqmhysu\nPYl4WcBYxcBY2ZIe/bDcePkWaAqwr4f+WTdsjFX4yZD06B2UAgM1MV7Hodl
612ufOlnF9GQVkxNV\nPH+uBsvlfV3yPPq8cYltK/IcNNVJeKTCUAyaW7/csaBpos+40Bfd1DZRMz
MgJt2kslAs10Ot4yYN\n/QAavaqIlYUXBvrE0BFxk0k30xqEjs1vForCguKw3vs5U+1AVXxUSibKJbdv
bvhKqLajPjeC6YkG\nyiUTN1zzDDTNw/QkP7cOnOs+x4rQsrxEAkC8G2g9lULbj6hHS4ZHH2Te5LV
CiG+vKD5KuouFIVdL\nhuPBdhkmJ/jNb9CAbGTohzeedcMOb3KAKHjMP7cs1wcDYh69F66sgKgP
UDoAzp/rXJB2HHlsaENf\n0VVctUPB3pPV3G3qpouxcrehj+tzk+MNzDXsRMOmluXixdkmtk1XMT
3RwOkII/Re8jT6POkm7jmk\nq1iFrDDIHFCAG2s9NPTZwU4u70QnnTgB05k38w1+DKLYJ/5Z+hnV
pumG+rGi+KGBj3vh/BiHb1Xc\nsb1wFaVpLjqW3zOL4Uy1g/GKASI+r7boUPQ8XM/PlYuWWjbKJTv
hwZVLFt747fdgxzbefnu80kZJ\nc/H8EIbe9xk6th9KfwAS4wTTq7Z+xHvl1A0nlN34cfLrJK+5WXr1
UNaHr44V5+D0BNfY89qQ1A0H\n//7Dj+BEqoWyMPS1zvCD55spj15VsyvbBYadOqcVD2aiFUX+6q
qyCl1NB2FDG3oAeMVuYP/ZemZ6\npO8ztE0vXEo3E9JN5GVPjXP9PV4h+8zpZfgM2DpdxdREHQ
zA08HKoVd6JdDtWTeNSGPlRVXRsQpv\nc9COlvwiFTePbJml2ragqdGFmCfdRKmV0cVWNCWSr
1ZETQL30hhjMY8+So0c9oI0Hd5ICuDfH0N3\nhlGcU0ttlMs8qKjrRpgZMwiO5+Nr+2fw259/Frf8yT
247f3fxNEM6YWP6uu9fyJgcqI2VEBW/E3j\n5xxvYxAFY0sF9Xkg6RQ0DCeRrTNeaUFXPdz/YnYj2
vQKQNc7YTbaoIi0ULHaySuaOjbfwtOna3j6\nVHLVHnfKhh3GzoeOJG+gvQL9nZTzwj36ZIUykO3Rl
8sm5hrmyKbLDcqGN/TX7CY4HguDpXFElsBY\naOiTPT5Cj34iMPSz0T6eOrkMAsPWyWo49PmpY
OWQ1+umSDBWTAgSHuli0NN7EG/X9Xy0LD/y6HPS\nF6ttK5H2x7vu+V3SzWxYLBXX6It69NFNT
Bh1y/VjHr0w+MXaHmfRsb1YjCMogMtJsWSM4VS1HVZ6\nlnR7qH43f/3No3jHPz6Nrz5/EhOTJ+
HBxK//w96uv+1iy4Su99//9EQNL842By6Lb1nJzw2l71Kk\nSHa3P+iFFkvP5O0MkufHnl2n8JX95z
M99bRMVNItzBeYSpWFMIqT4w2oih+2rEgjPn+66K1uRH+H\nmSEmejmeD9NhCclLVb2ecaR4mi+
QjEkBydqWNJWSAdNhYZHahWbDG/pX7Obrzn0nu3V6EXwt6RY0\n1Uvk1Ddi+lylZKKkuYnMm6d
OLGF6sglN81AumSjrTpiFk9frJk/uaFleeKFqWtIjFd7mIIY+XroN\nxNMrU1k3htN10pVLTpd0M5cql
op/lv4efTKvGOAeuKiQVULvJ7uYrAidoL9I/LjyPK+64aBt+aFc\nV9Lz+/v04uFjC9g6tYzv+fZv4Kbrns
GN1z2FE4tt/OcvPpfwyhZbVjiEohfTk3U4HsPRucGqJOMN\nzQS8pzz/TNU2nzdQlPhKrW46Xa992
aUn4HgMn83IEKmlZKJyyRo6/lGNZSuNV4xc6UaswrsNfcyj\nH0KnD3vRJ4KxvRv5xVtxAMmYFD9
GB7rqZbYwF6rCTGNtArIb3tBvHSdMVAzsO9Wt08eXmrrmdQVj\n4zm005NV3PXcObxwvgHH8/H
smRq2TC2Gv5+cqIXSTj+PPn6yiOlScelGbMMYC4wuC3Obi5DOj4/S\nK6Pj8nyGpuF1GXpds8Jls2C
2bkJV/IR3E46d62Oc45XCwqgbjgcruADiGv2gcQiB4biRoQ9WRnlF\nU6KcXujNJd1Gy0yW6t//4jwe
ObaY+36O5+OFmQa2TFWhBIPTtm9ZwrVXHcLdB2bDaU7i7xfPuMlD\naNGD6vTxoSOCpEZvDeTR
x9Mrax0rEYwEgInxNnZsm8enHzvRlQ7K5cLoOMq6BcPJHrbTj0QGT6mV\nm0svbnRp56RuOEHT
NjZUQDadcQTkV7YLojTfyImx3Liht3NltErYvnptArIb3tADwPTUIp46Ve3S\nv6LKP7srLa9leomL54Z
rnoPLOviFjz2Guw/MwHD8sJgKANfpg933y7qJL++Fd6+FenpkqBqGC8+P\nToKi0kZ4kQQnlUI+iFh
CA28YDhiySuAtVNPSTYMPtU53itXU3oaesSBQGHr0lhAbSTfC+4sbp0Hp\nxMrO8+QxgUitHKtEHj
2QjEv80Veex7s+/0xu/5kjc03YLsOWyaSmfvXlx7Br+wzef/chPHB4Hm2b\nxySyNNk042Nt6Ko3cO
ZNnkcfavSxWE0R4jUNdcOBntGg72V7jmOx5eDrz88knk+vEEulZHXs337r\nKP78Gy8WOo54/v9Yp
RP+3dKIG3pabuRVrRbGyvaKPHpdTRp6w/ZydXQjnWAQxKSEDMtTnrPPhXLg\n0a9V5s2mMPRbp
5ax1HK6miNFHr0LVbXDP67r+TAcP7FsG6sYeP0ND6NtG3jX554N9ps09AItR7oR\n1XVxDyc+75P
/z39uW26YQz8W6MlF5ZuoF33USEtP6YvpzpWCkm53ZaHMNkzoeveFlv4saQyH9/RP\ne/RcukkG
Y7VU1o1he7m6bJqOFXUX1Ppo9KGhLwuNnn9WIZGZjoczVV65+q2coOP+s/xvPZ0y9ETA\njdc+g
8mJBn7jM/vw4JGFxHv0QqwKD5wbLCAbDR2JGaSgMC2vc2Uv4oa+EbQoTrNj6zwmx9r4xCMn\n
Es/XjaRMFObStyycXGzjL795BJ989EShlely20ZJc6EQw1i5g6bpZRY+htJNO+3R82MplzpDGfqw\n
RXHM2dNU7njZOQ5At0YfODaBTJlOfohT1i0QmPToV8LWwPPedzop3zRiHr2qOqGBbKY0bsHEeB
u3\n3PAINM3FRMVlpBtOxwx9nnQDdBvcZsoji3v9IodeeJ9Fl8DRsjfujfiJKtYw1UtPSzfdw0dmap1E
\nxk20T7fnzScKFHZr9FZXemVyX3/3wDH86IceKpSFkOXRx/sFxT3z09UOKiUnLOIS3rbw6F9aaIU
D\nZ+54IrtScf/ZGkq6GwZ042iai5uvfwykdvDOzz0TvEcxjXx6soZDM42BOlmGHn1cugkC+i2LG6ai\
nfW6ASLqpdWxYLuuSbgB+U7piz0t49kwdz56JbkzL7eSAk3h17F/eewSeD3Rsv1Bh2HIsW0ic/2cz\
nuliKv3O1nTSQtWDYSqXUwZnlwStrQwdMS95AgfwEBMOJJBsAUIJzUVTHppMf4igKQ6XsYHaEc
woG\nocjM2CuJ6H4ieoGIDhLRO4PntxPRvUR0NPh/W/A8EdHfENExItpPRLes9oeYGm9AVz3sS6d
gxTR6\nTXXCu3gz4+IRTE82cNtND+B1r3488fzEWCu8g+dJN+J3cf04PgYu/n/TdMMceqEnF/Xos/
qbcH3R\njW2TnQGgazYahhd6XYwxzDftRMaNoN/c2FYqoBUfr2amCqbSvepfnG2ibhQL0BpO1HYi
vFEG7103\nHNzyJ/filz/xJGbrJk5X26iUo4CnMMKir4poGbvrkvN48MhCZqHOs2eWMTWx3CVlCS
plCze/+lGQ\nErV/LsL0RB2Wy3BsoXhANj4cQ6AqPO4jHlVBNHoee2GhhJAn+1y+8wxUxcfX9p8Pn
6ulMnxKQbbR\nQ0cXcNdz53HpDt5gLH0dZlFt29DE+MUgcJ71t2iGGn3yMwrJqlLm/agGTVsMWz
SngrFAvsMlrs+0\nVCluAFnJD3HKpc669uhdAO9mjN0A4DYA7yCiGwC8F8B9jLFrAdwX/AwAPwTg
2uDf2wH83ciPOgUR\nMD1VxVMnkh593XCC8nuux4uTRlw8WfokAIyPdTA5nvRKiCL5Jq9gCuguu

kgbw/jJtBh4meJEL+zR\nGzyAm/DoU7NAhUefzggp6TYYotVOrePA8RjK5SxD37uAJIw/pNMrHT+
zYCreSlk0z0oHhrMwHC8m\nASU1+oPn6mgYLu4/PI83ffB+PHemhko58vBCQx8YxSNzTSjE8KqrX
gDA8PnUHADT8XBkroXpyd7G\nanK8hZuvfxw7t890nSt5CCnowNniOn3aUeCP+Xch8scH0ejF60X
tRN5rNc3D1EQ9kfuflnpKug0i\nhs89dQa65uKGa/ajUrKLGfpYZbfw6LO6WIrrp5ZKS2waLrTA0A8zj
L2R5dH3GT4iri9FSZ7bpuPl\nJj/EKZU6OL9GRVN9DT1jbIYx9nTwuAngEIDLAdwO4FPBZp8C8BPB
49sBfJpxHgewlYj2jPzIU2yd\nWsKRuVaiKEpkCXAN2wlPmqw/chGmJmpQlez0KYGS6pfeSmVNiP
+5dBNo9OXBPPp6xw7bq0bvm+PR\n62lDz99TGAlxwVcypJt+6WYiuB3WCChRVWy6BUK8lTJjLN
TS+w0FSfcXUhTeN0d8ry/M8NqH77zx\nYZTLC2hZXkJy0VTeInoxNPQtTIy1MT7WwY5t8/jsk6cSu
e0vzDTg+egKxGaxdXoZt7z6qdzeR2km\nxlpQFH8gj75lutDU5N9afJcif3wQjR7gN13h0ecFDwF+vh
84V4fvs2gsZeyaIQIqJQeMAVddfgS6\n5vDEiJP5vacE1XaUlqprPC0xK2YjrmfbjZINnCDGxvvo8/N2
UJ2+kRnk7h3oD4OxatLQG7aHppmd\n/BCnUjLD6+1CM5BGT0RXA7gZwBMAdjPGRFh+FsDu4PHl
AOJu0tngufS+3k5Ee4lo78LCwoCH3c3W\n6WUwING2tGFGHoimObBcBtv1M3Noi/CKK47g5lc/0X
MbPmUqVoGb8sjSGn1Zd8MbTtGmX7VURaPY\nb/wEXWhZQdO25D5FJtFDR3l6YXrgSHqfvY5Je
PRqyqM33SyNProo5psWLJffLNNVumeqHfz+vxwI\nja/jMd5fKLaK0mOrsxdmGhgr2dg6vYxbX/MI
bnn147j68mh8GxGvHRAB6MOzdYyP8ZvDFbtPYrHl\n4L5DUVB2f6BJFzH0g0IETFQMnFwsrimL7q
BxxHcpjNsg0o14fT/pBuDfQcf2cWKpnRhSEqekG6iU\nbLxsDw/cbptaxtlls2/FbL3jJqdyVTqZufTxA
K1wCuKSbDSMfTBD3zST7R+AeGvufOlGUfww5Tae\nfLCcETdLUykZaFv57TRWk8KGnogmAdwJ4
F2MsUQZKuMC2UAiGWPsI4yxWxljt+7cuXOQl2Yi8pTj\ngaB45V+kjTuxIO1g6X6VsoVLtubnXwPdB
reZ0ljFAJKW5QZDpa2oOKlgT/r5phV2rgzfV3ETeepn\nlw2Ml80unblSNjE92cB9h+YAxEcIZhn63t3
8RNFSpFmKEz+eXpksMmlbbmJcXbpl8jcPzeEzj5/G\n8QVuDKNMh2R2RDsm3UyMc6mACNi5fT4
j08jCUsuG6Xg4u2yGUsuO7fMYK1v4+MPHQ413/7k6KiU7\n88Y3CirlJo4P4NE3Myaaie/ifAFjnYWq
RMNv8uRLIJKanj9Xj6UqJ7d/1cv347XXPx4aSZGplm5Z\nEMd0eKFRXFYsl9o4Xe3+XhqmA4X4vt
OGXkg3AHBuwOrYrIyjqIVJnnTjQov1FBI3XNPxY+miPTz6\nQB6dWwOvvpChJyId3MjfwRj7UvD0nJ
Bkgv+FW3QOwJWxl18RPLeqlHQHZd3F8Zi3VOtYUddILQrE\nDuvRF0FTXbRixrFlukHnRX6C8EZbv
MhkqcUDUoOM2nM9H88FwcLk+ybTF88ud1AuZXuOO7bOYN+p\nZdQ6dm+PXumXdZOt0Rs2l24
IfIAykEzrizfNSnv0QksXF4PoDqjEPHo+fMSF7fp4aaGFqYnu9hdx\nNM3EQssMM26EoVeI4erLD+P
Jk8v4p71nAYhAbDU3ELtSxsfaOF3tFB4t1zJdKGryOxI31Jn6cB69\nosQCkD1uEhPjLWiKj/1n6121
G4Jt08vYOhWtfqYn61AVv6dOX8vICBurdHBm2egKqrYsN9TwxeuS\nhZC8WndQj365Y3fJVlrMGc
mCJwVE52G8QLBX+wOBkEfXIiBbJOuGAHwMwCHG2Adjv7oLwNuCx28D\n8OXY878YZN/cBqAe
k3hWlbFKM9HlTqRgAQiHKzdNNzOHdlRoqpvwzOODOeLbtEwXCy0TJd3KrGzN\n49BMEx3bx7bp
pA6aTl88U22Hy9o0O7fPw2d8nml6hGDys3iZoxHDz5ZOHY1JN6IRmfjc8Yyc00sd\nEDEQWFeqp8i
OCQ19KneZP7bRNB0cnW/C9ZM1Dllwj94K2w/Eg6dXXnoS27cs4b999SCOzbdwYqHT\nlT8/SsYr
bVguw1zB/juLLRN6Kjc7Lt3kldz3QknIYPmGXiGGySAgm55Glb9vH9OTtbAvVBbVjDm3\nY5UODNv
vWuG1zMjQZ0k34TD2AQ39TL2TGLQD9G/kF0/zBZLBWJHn3+v7FAkP69LQA3gDgLcC+D4i\nejb4
98MAPgDg+4noKIA3Bz8DwN0AjgM4BuB/A/jN0R92NuOVFo4vRBdxM6XR8+f4zFFd9UKtbZSo\nq
gvLjbJLeDAtvfR2wmBsSbfCytYiHv0TJ7iBj1ftAkjMAjUdD9W2i7FK9sm/ZXIZZd3Btw7NY7Zu\nZub
Qi8/iMyTKvOO0LD63NupnE6+M9cPgLBB5S8Kjn6iYKOluV5Wu8Ojng2pL8Zm02AWmqfxmLcY7\n
9vPoeZEYvzEoxDBeiSQCPgHoWRiOg1/+xJNgALZMraKhD3q+n1wsln1xvmaE8oQgHowtOlkq8fo
w\n1bf/TWJ6chnPn69HQ8gLvN+WqSqeP1fvGrMnSA+tB6LGg/FOlI7nw3JZmKwgdPBG6qZTKrU
H7mc/\nUze7VrFaH4fLdJLT5dSYR9+rc6VAyKNrUR1bJOvmYcYYMcZuYoy9Lvh3N2NsiTH2JsbYt
YyxNzPG\nqsH2jDH2DsbYNYyxGxlje1f/Y3DGx9qYa9hhGXPDdGMafXCSmHwU3Wp48/x9kicL11i
TF4eiOqgZ\nDuqGh5JuRZWtBTT6J09UMTHWSRRzAfzidX0+GFt4N2M5Hj0RcMm2Gdx/eA5nlzso
ZVTFin0C+UvZ\ntuUlVitizJ3lBB59hp7ZsVycWGyhXG6ipNtdPUxEBau4GMIiFTVp6Fumg0MzTaiKj
4mx3pp3KejJ\n8tyZOibG2l3GbWKsjVde9UKY3rcagdjwvYKbTF7P9zg8yOdm/K2jc2tQ2QaIbpqlAtf
A9GQdhu3j\nmTNciinyftumqnD9/EErWX3bRavneAWsWDGGHn07rdHz31fKg/V6F4N20t8rd1hYzz
x6ilXFR+nE\nHmod7vT0koMVxUel5Kxbj37DIDy1U9U2OrYHz48km6gZlpNb+j0K0kUXTdPp0lg1xQ
lTyYQH0C/w\nCfD++k+cWMLWqe6AsHhfw/bC7lU86QYAdm6bQ8P0cHS+nRt4FAYhT6dP9/Pmx
+GH6ZUJXT3m0Z9a\n4m2EVbW7wdpCKxmwigaDJ9PgWpaLF2bqmJpo9NXTRUrp3lPVMOMmz
VV7jmPbdBWTY+1CvWuGpVI2\noCh+IUMfto/O8egBDNS5MnpNsv9SL8RNj2dpsa7VaeZrgoBsnk
5fzfB+o6E40XMiqaGk29DVyGuu\np24UY2UD1babu4JIIwbtpFeyUfwsez9t201990mPvqS7fc/Fcq
mzJtWxm8rQTwTL4hML7URDMyDy\n6JuBR68oq3Mxq7FYAP/f6brLa6obBkGFEdL6dM4DgGMLL
dQNt0ufB2Ll27YbdvPLk24AYMfWhVC6\nSns26c+Sl2LZzsoICfqwmLEiJ76vKIDYsnyMV9rQdSvU5
AVCv50N2rmm+4sA3EC1LA8Hz9cxOd7f\n+xZGxHT83OImIuCWGx7Dra95uO/+VsIgKZZCxkjXO
MS/iyItktNEbXb73yQmxprQVO48lFO1G3mU\nSzYmxzq5hr6WodGLx3FDH68K1nUnEYyNy06VW
MOwtuXiNz6zD1/cdzb3+KLake7zvtfcWMN2MzV6\nEVso4iCsVdHUpjL0wqM/sdSOql+15DKvabq

opWZFjhLRE0c0zGqYDtItE1TVDTthCkOvKG5XP/k0\nTwSVv9u2ZBj6WP/4s8s82NkrRVDTohtGL4
2e7zPHo7e6b5iKwvV5M8ejPxQMYefj9Rwsxwy96Xho\nW1zuER694SQDvUDUfKppen31eSDZi6Z
XFaumemH/ltWkaIqlKljq8uhjhn44jV549P0NU7wifJBV\n8PTUIp48sZSZXVTt2NC1ZHxAfI54FlY82
K9rViIYG2+XXClxw3l0voW3ffwJfP352bDhXBZipZRX\nO5JXMNWx3cR3TxSc7y736IvcOCslEzPrN
b1yo6BpHiolm3v0qQIGhRg0lVewpae/j5KJ8Ta2b1nC\nZ544Cd9nfLpUytDH5Q7hBSiK07dg6qkTV
YyVrUztPZ6+eG6ZT53vF2zesW0WQLZnA8Skm5ylbHzo\niEBR3LCpGVH3MvdQUMk6PtaBrtuoG
W6Y1RMNo7Cw0LSDeand0k38++yXccP3FxmPou0KVpOiKZYz\nOR59PF13GI1efJdFDfdUUKOi9h
mZGGf79BLqhosj893fd63TXfCnEENJdxNSXjwNWovN/q2nWjOL\nQO5vf+FZ7Du9zFuS9yhKiiSxr
JRiJ9exiU86E2gKH3pfbRebC1ApG2gY3tBjNYdlUxl6IEqxzCrw\n0DXeCrVlugmPYNRcsfsEzi2beOj
YltqW17V6iBuquHST50kAvBXA48cXsWVqIXP5HB/9d2a5g3Kp\nv8d42a6zuPLSk2H3z6599pFu4
mMEBYrCG5rxYSFRMFYEak8E2vRYuQ1ds3lpe6CtioybqYk6fMbn\nsWZKN3FDP17co09n3KwVRV
Msz9X4ZDMxLzeOrq7A0AffZa9iqThCp9f6pFbG2b6Fx5Eef6l79Rlv\naBanpNmJaWDx9iG6ZieCsX
HvWRjsluXgpuv2YnqyltnyWDDbMKGpXqaz16u/E88kS1cp+7ACj75I\nKwqxirjQrRA2naEfr7RwfLH
ZNcgYQHCn5wVTq+XRA8DuS2ZQ1m188pETsN3uAFZYYERR7xC1T+/3\nM1UD800b2zP0eb7PK
Af47HIn9HJ6UdJt3HDN/txeLf3yirNWK0Tco+dZN8n9aqoHxoDxsgVV9WMB\nOP4diP78wkufa5g
9Df3kWCdsR9wLflF7mRk3a4HIEoqnWH51/3kcPJ9cncxkpFYKQmM9hAQ5SDAW\niCpkB7mpjFU
MjFdMPH48y9BbmbKRlpp+Fp/lUNJtLIeN+JKvVxQfr7jiCF53/ZO4dMcMT781ehv6\nSqI70I54r2bG
JDTGGDf0qWtFpDWnVxl5RJOmLmxAdvMZ+rE2qm0X50XDp0R3OhsLLQuuP3hDs0FQ\nFlbLdp3
C/Ye5TphXwl4uOVFBUZ8JTGH+fIY+H99n3XCw0LTDlLSVkDcDF+AZQNWW3dWLXWQPpYOx\nf
H/853KZL+fDAFxwcQuPfjrQ3eebJjpOsr8Ifw/+t5soEIgVlEvGQNuvJqLpmsi8WWpZeOfnnsHf\nfut
YYrtztfzq5kh+WX3pZmKsBV1zBo5fbJ2ex2PHu3X6eEOzOHpMngHiYxS5EW2ZHlzPz5yqde1V\nL
2LX9rlw+yxjLZitGyiV8lKKsx0u2/O7ei4BfAVbMxyYDisUGK+s0aSpTWfoReaNaK+a6OOtOjgX\npB4
WSRNbCVdceip83J2CKFYb0Undb8jHkyeqKOtObs64MPSizL9XamVRopTI7u+q2rHh+tGINIGi\neD
AdF6abzKOPH6MwdOmUOtHNM/LoLRi2l+gvAkTfZxF9XnDz9U/g+pcfLLz9apJOsbzrufPwfODQ\n
bMqjr+d79GKG7kqCsUUNPRHwHTc+hFdccWSg99m+JVunrxlOpsyh63YiC6tpumFvKHFDqxsOb1
bY\nY0UuKs/zmKkbuYkK/Drsfm1WzyWA/x3CBnEDSDcXOpd+0xl6ocE+e6aGkpbMa9U1J8zmWK
08+ug4\nOtixlWfeZKVXAoCuG7Hn8tsNGLaHew/NYuv0fG56m/A0jgRl/kWkm34Ig5Dl0c/mNENTl
SCP3vG7\nPHphnER1qLh4RRB2qW1DU/zwRjBb59JNWloar7QxNVHDzm1zhT/LxHj7gmTUFCGd
YvnFfbzZ6+kl\nI8wFb5oOWpafGygX/XQ0j/EAABozSURBVGqGKphSB18NTI63cgdf57F9ulunt10fb
cvPfO+SZida\nIMQ7TAojutiyedvqHtevpjpoW35msNv3GeYbVmZbboBXcGed78IJS5/TRG5otIt8n5
rqYaxs4dOP\nncBDR1fetbcom9DQB8vhtt0VcNVUJxwjt5oaveDKoHVrXql1PBukV7uBzz11GrWOi6
suO9H1u/Q+\njwbdO/OqYgdBzMDN8nDyDL0STJKyXJaxzOXfedqjFxf3YstCuWTzsWslJ5JuUoZe
11x81+sexPRk\n/0DsekWkWB6ebeLg+Sa2TlXhM74iAyKPbzU0+i1Ty3jVy5/v24l1pXCd3sBjMZ2+
ZuS3CtB1Pt5Q\neM+tWEGekEVOLQknoYeh11wwZCcRhCvRnBtoqWShY/tdbZbTvegFqurltnDO47
WvegJtdxlv/diT\n+C9f2t8zCWNUbDpDr6o+xivZQxUSg4BXMetGsGv7HN5w87e6SupVcfLqSekG6
G43YLs+/te/HsO2\n6WpXf5s4fEScj1PVDgCWayAGRVP9TElpJux6ma7a5IFY2+326IXhF6MThVd
WjWn0WjCerlQyAunG\n7Upp2wyIFMs7nz4LhRiuveoQgKjNtmhjkWvoFWHoB/foFWK4+rLjiayo1W
Lb9AIeO74YetdZYzAF\npVTRVDxpQg8NfSf4ubdHL16fpldqJQDs3s77L375mfOJ50PpJuecBlB4xb
NlqobbbrofV19+DJ97\n8jT+n398utDrVsKmM/QA95aA7sq/uBe/2tKNYHK81SW3RB59dHx57Qbu
eu48Zhs2Xn55f31UU30w\nBoyV7ZFll+TFDubqZlCUle6L76Fj88BV2pCIm9lY4NGL3GnR5GqxZaIU
pN2VdV4qzoc9bEJDH6RY\n3vH4KVyybQ5bp6pQyMfh2aIePc94Wg9ZRL3YtmURDcPD4eAGFtVK
ZHv08W3iBXkiFnFyKSn7ZaHH\nprilmctxUAQT421snarhn/adTsioWV1UgXQn0OI3XVX18aqrX8D3
vv4Z/M4PvKrw64ZlUxp6EZDt\nyl9PpVquFWXdxNRELRzSAGTnrPs+w4cfOIrpiSZ2bJvv2k+aMKsl
J1NjGPhAk+4LZqZuYqxkd93E\nFMUL5bG091PSbVRKRkJSi+dOL7as0ACUSyZmGwY6trspDb0Iqr
dtD5ftPANFYZgcb+PILJejZmoG\nCPnVzbu2z+KKS09eqMMdGpEOLNIse/VtTwfn+eAgfm4Ib18Y+
l4avRp69N3bzPQYtCPYs/M0jsy1\ncfB8JA2G0k3qXExMPhuiR9llWxp4zeVbBn7doGxKQy8CsunIv
B6LmK+loVdVH9/1ugexfUuGoY8F\ngu55YQ7HFzq4+vIjhXqMiH2MIuNGoORUCs42slPU4pk2aY/
+misP49bXPJp4TuROM8ZQbTvhKqdc\nsrDc5u2Iew1j36iEcQrNDdMCx8fqOBQY+vN1E5WynVvdv
PuSWbzq6hcuzMGugLGKgYmKgYeC2Qdi\nVnBm1k0qON+KFeSpwexfUXvQa0UurvNGhnQz1zBB
YCj1CMxfuuMcFMXHnU9H/XIMu7sVBxB59Jra\nnWW2ntichn4sO2AjjDuBdS3B1pqseZUffeglTFQ
M7N5xPu9lCYTnO4ocegHv/ZHlGRmZy99kf5u0\nR++Eqy2ByJ1uWi4cj8UMvQkG4Nyyue7+VqOgU
jagaw4u3Xk6vCFOjTcxU7fQNB3M1AyUc3K9Nxpb\np+fxrcML+M7334f33/0iN7S9PPpQuokK8oiA
su7Gpmr1Csb21uh73UD5cTjYuW0W//LMWTieH1Sl\nV4N9p2tiird8Xku0tT6A1UAsi7ulG5HWWK
wL34Uk3sYX4JV4B2ca2HnJ+cIDUkRWyyhSK6Pjyi4g\nma2b2HFJ9pzZ6Hj6G2hdt1HtWGGxVFy6
AXgW0mYMxhIB/+a1D6QarnFv/uh8C2eW25vG0F971YvY\nOr0MxghgQKViZLZ14Ncrw3LHAWM

MbcvDJfEWJroNwy7Fts1G3ByyculnGyZKev/r47JdZ/DMocvw\nwOEF7D1ZxScfPYnLd5/qymYTzk
x6hvN6Y1Ma+vFKG5ftOt2la4uT40IFYgch3W6gYbowbB9jOdkB\nWWirIN3w9snJC6ZpOujYfmYuc
tKj77+U1TUHc8tOWCwlDF9cQ92MHj3Q3UZ6coIHLA/PNjHbsHDZ\nrgvft3w1KJcsXLH7dN/tFGIo
67xvjOn48PxkYSNPrpiEqvg9zy2tp0bfyQ3ExtmxdR6Vko33/NNz\nqBkOrrz0BF79igNdDqI4jvVu6Iv
MjP04Ec0T0fOx515LRI8R0QEi+goRTQfPl4joE8HzzxHRG1fx\n2HscM3Djtc9iy2Sy0lBo9mupz+eRb
jcQdi4cwDsXBnGkHn1G1k2YuZDZ/W8wj76kWzAdFvbQj0s3\n0T43n0efxVi5A0318MTxJdju6FJk
NxK6bqPatsMWBnFDLzT8kt77fFBVDwTWQ7rp/70qCsPuHWdQ\nMxxcteelTCPPtxs+zfVCUkSj/y
SAH0w991EA72WM3QjgnwG8J3j+PwJA8Pz3A/gfRLRu4gBCulmt\noSMrId5PHoh6kRfxPsJ9CI1+
xNJNuqXqbF143b09+kLSTbC6OjbP5TZh6PmIRS5ZbVaPPg0RMDHe\nxINBxWRervdmRlNNLLftq
EVxSrrh2/S+fvmkKK/Lo+/YLlqW33NOQ5xXvuxF3HLDY3jVyw/2rUgf\ntGr4QlNkZuyDANKVOtcBe
DB4fC+AtwSPbwDwreB18wBqAG4dyZGOgKj4Yv39UcIJUYFHf34Ij35q\nooHpyeWRGkYx7NyLlZN
HfdJ7e/RFpBuhyUeGnv9MBFRKdtc+NzuTYw1Ug7mpeWX6mxld4/1u4kNH\nBCKAqxbwnnXNDZ
uiCfKqufPQVA87t2W3BRdEFcrrz3mMM6y3fRDA7cHjnwJwZfD4OQA/TkQaEb0c\nwOtjv0tARG8n
or1EtHdh4cL0fOAI/d2tddcD6XYDMzUzyKMurv1dddkJ/JvXPjTS48rKBuot3cTT\nK4t49PwCOTrfQ
klPFgCJ1czF4tEDUUAWGOwmv1ko6Taqncij1xNFjkEOfoG++GrQkjzObFgsNbqV\n0maSbrL4FQC/
SUT7AEwBEJ/y4wDOAtgL4K8APAog8ypljH2EMXYrY+zWnTt3DnkYg7Nn5xns3F68\nGdaFRFO9c
G7s+brBK1wLZtysFlnjBGfqwUCMDI89Kd0U9+hPLrZRTuVWi+yli8qjDyZgKYrf1QL6\nYkDXeWOzs
Bd9LG1RSDdFVuSqandJN5FHP1ppM35s65Whsm4YYy8C+AEAIKLrAPxI8LwL4P8V2xHR\nowAG
6226ytxwzYG1PoRcVNUL58bO1s3cntkXkmhyVdKjz50z2yOPPgvhCbk+g6Yl9yk8r4slGAtE\nHv1Y
OXswxmanFEwdm2+KyWtO7HfdU+Py0FSna/jIbI+V6LCI87eoHLRWDOXRE9Gu4H8FwO8D+Pvg\n
53Eimggefz8AlzG2/sv31gmq4oYe/dnldjj0eC2JpJvIaJ+vGyjlGPrBg7GRJ5T2YENDfxFJN+WS\nhZ
LmoqSPro3FRkJ4xqKCdliPXlPdLo9+rm6ipLm5E9WGYWqigVu/7dFwdOJ6pa9HT0SfBfBGADuI\n
6CyAPwQwSUTvCDb5EoBPPBI93Afg/ROQDOAfgrSM/4k2MojjoWHxY9nrJo87qqjlTNzAx2bt9LlAs\
nGKsoDLrmwXHVLkNfuQgNPRHwssuODZRttZkQAVdh6NWM9MpCfd81F81Ot0Y/Sn0e4H+v1W7
3PAr6\nGnrG2M/m/OqvM7Y9CWD1W7FtUlTVRct2sdxx1k0etZry6G3Xx3LbxfbtORN6BgzGAlynd9
yxro6G\nW6ermJ5cXhcDvS8k11y5rtTOC0ro0S/xmoJ4jGq80sHObbPYnjNOM46mOmiZ3GmiQAM
rWhW7GdmU\nlbEbFVV10bacsBf5qL2PYUi3ZhAZN3k53nwxx6AqrLDGrKkWgLEuj35irD3yLCLJ+q
YUVJiernag\npwa/K4qPW254stB+NM2B6/MWGhVdBcBnxZYrF6ehXzfFTBLek75tuX17kV9ItFT75
NDQ50gLPE20\nd4l6Gl1Ptj+QXLwIj95wvBVVsIvzVqRY+j7DQtNeF87TWiAN/TpCtBsQBUmjrHAdFi
HdiAwGcRPq\ndcH060WSJiptl4b+YkcPGpsBUV/5YUj3u1lq2/DZ+s+OWS2koV9HcEPv43zNhELrI
49a12xMjHXw\nuadOw/X8mHSTfxNSVb9rzmsvSqGhX9+5yJLVR7QjBngu/LCIbB3h0c+tQrHURkI
a+nWEpnpwPIYz\nyx2MlbunN60FRMC1Vx3Esfk2vrD3LGbqJjTV61ldrCgeFCru0QsDvx5ubJK1R/S
NWUkFe3purBj2\nfbEaehmMXUeIwOdL8y2URjgOcKXs2j6D7dNV/MU9h3DTFdtQ6VPMo5AXBGW
Lcfnu0yiXjXXfGEpy\nYdA0E8D4yjT6cG4s38dco7sr6sWE9OjXESJn+PhCa101tCICrrv6eVTbLh44v
NC3mEdRXNAAbQvK\nJQuX7zrbf0PJRYGuiarY4T160SOnkZZuLtJVozT06whRsWd7DOV1EIiNs2
Wqhj07uDHuF9Dadcl5\n7No+cyEOS7IJEcH59Ni+QVDVtEZvoVJyEk3zLiakdLOOiFcBrofUyjTXXn0I
89U9GB/rXcD08stf\nukBHJNmMRH3nVy7diKyb+R79mS4GpKFfR8Sbd42tQy1xrGzgu2+5b9136p
NsbEoj8OgVYtBUL/To\nZxoG9Iu0KhaQhn5dEW+2tB49euDinHokubCURuDRA3zKVCum0VcmLt5
zV2r064j1Lt1IJBeCUKNf\n4YAgTXXQtBy4no9qy7loi6UAaejXFcLQq4q/7ifWSCSrxdbpZey+5DymJ
2sr2o+q2mgYLhZaFhgu\n3tRKQEo36woh3YyVzXVRLCWRrAUl3cbrrt+74v2oqoOGaV/0OfSA9OjX
FSIYux4mS0kkGx1N5SMJ\no0Z80tBL1gGKwqCQv66KpSSSjQqfMuVi/iLvcwNIQ7/uuGzXaey6RB
YbSSQrRdNctCwXcw0LROyi\n7qXU19AT0ceJaJ6Ino8991oieoyIDhDRV4hoOnheJ6JPBc8fIqL/sp
oHvxn5tlfux+5LZtf6MCSS\nDY+mOjAdhnM1A5XS+mgSuFYU8eg/CeAHU899FMB7GWM3AvhnA
O8Jnv8pAOXg+dcD+HUionRkRyqR\nSCQDIIaIv7TQyh1mf7HQ19Azxh4EUE09fR2AB4PH9wJ4i9g
cwAQRaQDGANgAGqM5VlIEIimOyMN/\naaGF8kVcFQsMr9EfBHB78PinAFwZPP4igDaAGQCnAfw
FYyx9kwAAENHbiWgvEe1dWFgY8jAkEokk\nG9FCoW15F3UgFhje0P8KgN8kon0ApsA9dwD4DgA
egMsAvBzAu4noFVk7Yx9lhD2K2Ps1p07dw55\nGBKJRJNvIXCxW7ohyqYYoy9COAHAICIrgPwI
8Gvfg7ANxhjDoB5IoEwK0Ajo/gWCUSiaQw0tBH\nDOXRE9Gu4H8FwO8D+PvgV6cBfF/wuwkAtw
F4ceWHKZFIJIMR734pDX0fiOizAB4D8CoiOktEvwrg\nZ4noCLgRPw/gE8Hm/xPAJBEdBPAUgE8w
xvavzqFLJBJJPvGmaJXSxZtDDxSQbhhjP5vzq7/O2LYF\nHpyVSCSSNUXTpHQjkE3NJBLpkRVfC
gKH1J/sXeDlYZeIpFsWnTVg6Z6F3VVLCANvUQi2cRomgtd\nu7iLpQBp6CUSySZmy+QiynJamzT0
EoIk83Ljdc+s9SGsC2SbYolEltknSEMvkUgkmxxp6CUSiWST\nIw29RCKRJNjjT0EolEs
smRhl4ikUg2OdLQSyQSSZHGnqJRCLZ5EhDL5FIJJscaegl\nEolkkyMNvUQikWxypKGXSCSSTS\nDgRzRPR
87HnXktEjxHRASL6ChFNB8//PBE9E/vxieim1R6ERCKRSPIZxKP/JIAfTj33MQDvZYzdCOCfAbwHA
BBj7O8YYzcxxm4CcAeAE4yxZ0Z4vBKJRCIpyLKGnjH2IIBq6unrADwYPL4XwFsyXvqzAD5/UUcnkUgk
kotmWI3+IID/EDz+KQBXZmzz0wA+O+T+JRKJRDIihjX0vwLgN4loH4ApAHb8l0T0BgAdxthzmS/m27ydiPYS
0d6FhYUhD0MikUgkyzGUoWeMvcgY+yHG2OvBvfZjqU1+Bst484yxjzLGbmWM3bpz585hDkMikUgkAzBU
rxsi2sUYmyciBcDvA/hw7HcKuJzzxpEcoUQikUguikHSKz8L4DEAryKis0T0qwB+loiOAHgRwHkAn4i95I0A
zjDGjq/GAUskEomkGMt69Iyxn8351V/nbH8/gNsu5qAkEolEMjrkBkMJpnRQyjk9mmS2UBi8bn4/Fa9Tr8z6y8/DfzYcX8nl+4e8jQinT4mTeXuo"

cWIpqLdNLEJJITQxHpJG4wYo/+ZmCompoRQaLVKjfQircYGsWb9A7EL\nRVjKSgEvbEtla29GTaH
684/zkozrjrOcndnpvvN8ksmc854zM7+Xd3ly5p0zcw7WsV4zM7tANYM+\nIrqAlwY0twNdaXk3cP
MgD+0EHhxWdWZmNmxl5+iPACvT8ipg5iD7fBrYUe0JJK2V1C2pu7+/v2QZ\nZmZWS9mgXw2sl7
QfmAL815V3JX0A+EdE9Az2YICI2BIRHRHR0dbWVrIMMzOrpdQVpiKiF1gGIKkd\nWDFgl1v4P0fzZ
mY2ckoFvaRpEXFG0hhgI7C5YtsY4FPAh+pTopmZDcdQTq/cCAQCQ6JR0\nDOgFnge2Vz
zkWuBURJxsRMFmZnZhah7R2buz/K2DJMGoyM7M68jdjzcwy56A3M8ucg97M8ucg97M\nLHMOe
jOzzDnozcwy56A3M8ucg97MLHMOejOzzDnozcwy56A3M8ucg97MLHMOejOzzDnozcwy56A3\n
M8ucg97MLHMOejOzzDnozcwyN5RLCW6TdEZST0XbQkl7JR2W9JikSyq2vS9tO5K2X9yo4s3MrL
ah\nHNHfBywf0LYV2BARC4BHgDsBJI0DHgDWRcR7gOuAc/Uq1szMLlzNoI+ILuClAc3tQFda3g3c
nJaX\nAYci4nfpsX+NiUVqVYzMyuh7Bz9EWBlWl4FzEzL7UBIekLSAUl3Vq7u/vL1m
GmZnV\nUjboVwPrJe0HpgBnU/s44Brg1nT/CUk3DPYEEbElljoioqOtra1kGWZmVsu4Mg+KiF6KaR
oktQMr\n0qY+oCsiXkzbfgYsBvYMv1QzMyuj1BG9pGnpfgywEdicNj0BLJA0MX0w+2Hg98Mv1czMyqrkJ
m5Qzl9Mod\nwF5gnqQ+SWuATknmqQ+SWuATknHgF7geWA7QES8DNwNPAUcBA5ExE8bVbyZmdVWc+omljqrb
NpUZf8HKE6xNDOz\nNwF/M9bMLHMOejOzzDnozcwy56A3M8ucg97MLHMOejOzzDnozcwy56A
3M8ucg97MLHMOejOzzDno\nzcwy56A3M8ucg97MLHMOejOzzDnozcwy56A3M8ucg97MLHNDu
ZTgNklnJPVUtC2UtFfSYUmPSbok\ntc+W9E9JB9Ntc/VnNjOzkTCUI/r7gOUD2rYCGyJiAfAIcGfFth
MRcVW6ratPmWZmVtZQrhnbJWn2\ngOZ2oCst7waeAL5S18qGaPyY8UyeMLkZwak
depGfRVHAFWAo8Cq4CZFdvmSHoaeA3Y\nGBG/HuwJJK0F1gLMmjWrZBmwdM5Sls5ZWvrxZm
a5K/th7GpgvaT9wBTgbGo/DcyKiEXAF4Afnp+/\nHygitkRER0R0tLW1lSzDzMxqKXVEHxG9wDIAS
e3AitT+OvB6Wt4v6QFNE93Xao1M7MLVuqIXtK0\ndD8G2AhsTuttksam5bnSuBucDJ+pRqZmZZl1Dy
il7QDuA64XFIf8FVgsqTb0y4PA9vT8vT8vT8rXA1yWdA/4N\nrIull+petZmZDdlQzrrrprLJp0yD7PgQ8NNyiz
MysfvzNWDOzzDnozcwy56A3M8ucg97MLHOKiGbX\ngKR+4E/DeIrLgRfrVM5o0Yp9htbst/vcOi60
3++MiJrfOH1TBP1wSeqOi0iI5m1zGSWrHP0Jr9dp9b\nR6P67akbM7PMOejNzDKXS9BvaXYBTdCKf
YbW7Lf73Doa0u8sxujNzKy6XI7ozcysCge9mVnmHPRmZplz0JuZZc5Bb2aWuf8CiHH5GZ\nJBRVRTR
d6T2yyTtlvRsup/a7FobQdJYSU9Lejytz5H0pVkk7JfVKOirpg60w1pl+n/6+ey
TtkHRxjmMtaZukM5J6KtoGHV8VvpP6f0jqD\nPv3u/XeBG4H5QKek+c2tqiHeAL4YEfOBDBJ
cDtqZ8bgD0RMRfYk9ZzdAdAwtGL9G8C3IuJdwMvAwtGL9ZXUnFmP0Jr9dp9b\nR6P67akbM7PMOejNzDKXS9BvaXYBTdCKf
MR7gbHALeQ51vcBywe0VRvfGymu6e4XfGymu6TGX4rKr95R90VEb9MD7\ngeMRcTIizgIPcTou
v1H8x59O0df70273Ax9vToWNI2kGxdXLtqZ1AdcO9MuWfVb\n0lsprulwL0BEnI2lV2iBsab4yfS3S
BoHTKS4LGl2Yx0RXcDAa3RUG9+VwPei8BvgUklvL/O6ozno\npwOnKtb7Ulu2JM0GFgH7gCsi4nT
a9AJwRZPKaqRvA3dRXMQG4G3AKxHxRlrPbcznAP3A9jRdtVXS\nJDIf64h4Dvgm8GeKgH8V2E/e
Y12p2vjWLeNGc9C3FEmTgZ/Dn7C3FEmTKS7q8rmIeK1yWxTnyGZ1nqnIjJ/\ns2sZQeOAxcA9EbEI+DsDp
mkyHeupFEevc4B3AJP43+mNltCo8R3NQf8cML/iW/mruDG4/iG4rHrt\n0/fFDA3fTI/geLDm11Nrqnu0rz0vcDRiLi7YtMu4La0fBvwk5GurZEi\n4ksRMSMiZlOM7S8j4lbgSeCT
abes+h0RLwCnJM1LTTcAz5D5WFNM2SyRNDH9vZ/vd7ZjPUC18d0F\nfCadfbMEeoqpAE8CXm11Pg/p4DcVbuUPAwXS7iWK+eg/wLPAL4LJm19rAf4Pr\ngMfT8pXAb4HjwI+Bi5
pdX537ehXQncb7UWBqK4w18DWgF+gBvg9clONYAzsoPoc4R/EOb218QVE\ncWbhCeAwxVlJp
V7XP4FgZpa50Tx1Y2ZmQ+CgNzPLnIPezCxzDnozs8w56M3MMuegNzPLnIPezCxz\n/wEY0siNlc
kV2gAAAABJRU5ErkJggg==\n",
"text/plain": [
"<Figure size 432x288 with 1 Axes>"
]
},
"metadata": {
"tags": []
}
}
]
},
{
"cell_type": "markdown",
"metadata": {
"id": "4_kCnsPUqS6o"
},

```
    "source": [
      "You can import your own data into Colab notebooks from your Google Drive account, including from spreadsheets, as well as from Github and many other sources. To learn more about importing data, and how Colab can be used for data science, see the links below under [Working with Data](#working-with-data)."
    ]
  },
  {
    "cell_type": "markdown",
    "metadata": {
      "id": "OwuxHmxllTwN"
    },
    "source": [
      "## Machine learning\n",
      "\n",
      "With Colab you can import an image dataset, train an image classifier on it, and evaluate the model, all in just [a few lines of code](https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/quickstart/beginner.ipynb). Colab notebooks execute code on Google's cloud servers, meaning you can leverage the power of Google hardware, including [GPUs and TPUs](#using-accelerated-hardware), regardless of the power of your machine. All you need is a browser."
    ]
  },
  {
    "cell_type": "markdown",
    "metadata": {
      "id": "ufxBm1yRnruN"
    },
    "source": [
      "Colab is used extensively in the machine learning community with applications including:\n",
      "- Getting started with TensorFlow\n",
      "- Developing and training neural networks\n",
      "- Experimenting with TPUs\n",
      "- Disseminating AI research\n",
      "- Creating tutorials\n",
      "\n",
      "To see sample Colab notebooks that demonstrate machine learning applications, see the [machine learning examples](#machine-learning-examples) below."
    ]
  },
  {
    "cell_type": "markdown",
    "metadata": {
      "id": "-Rh3-Vt9Nev9"
    },
    "source": [
      "## More Resources\n",
      "\n",
      "### Working with Notebooks in Colab\n",
      "- [Overview of Colaboratory](/notebooks/basic_features_overview.ipynb)\n",
      "- [Guide to Markdown](/notebooks/markdown_guide.ipynb)\n",
      "- [Importing libraries and installing dependencies](/notebooks/snippets/importing_libraries.ipynb)\n",
      "- [Saving and loading notebooks in
```

GitHub](https://colab.research.google.com/github/googlecolab/colabtools/blob/master/notebooks/colab-github-demo.ipynb)\n",
"- [Interactive forms](/notebooks/forms.ipynb)\n",
"- [Interactive widgets](/notebooks/widgets.ipynb)\n",
"- <img src=\"/img/new.png\" height=\"20px\" align=\"left\" hspace=\"4px\" alt=\"New\"></img>\n",
" [TensorFlow 2 in Colab](/notebooks/tensorflow_version.ipynb)\n",
"\n",
"\n",
"### Working with Data\n",
"- [Loading data: Drive, Sheets, and Google Cloud Storage](/notebooks/io.ipynb) \n",
"- [Charts: visualizing data](/notebooks/charts.ipynb)\n",
"- [Getting started with BigQuery](/notebooks/bigquery.ipynb)\n",
"\n",
"### Machine Learning Crash Course\n",
"These are a few of the notebooks from Google's online Machine Learning course. See the [full course website](https://developers.google.com/machine-learning/crash-course/) for more.\n",
"- [Intro to Pandas](/notebooks/mlcc/intro_to_pandas.ipynb)\n",
"- [Tensorflow concepts](/notebooks/mlcc/tensorflow_programming_concepts.ipynb)\n",
"- [First steps with TensorFlow](/notebooks/mlcc/first_steps_with_tensor_flow.ipynb)\n",
"- [Intro to neural nets](/notebooks/mlcc/intro_to_neural_nets.ipynb)\n",
"- [Intro to sparse data and embeddings](/notebooks/mlcc/intro_to_sparse_data_and_embeddings.ipynb)\n",
"\n",
"\n",
"### Using Accelerated Hardware\n",
"- [TensorFlow with GPUs](/notebooks/gpu.ipynb)\n",
"- [TensorFlow with TPUs](/notebooks/tpu.ipynb)"
]
},
{
"cell_type": "markdown",
"metadata": {
"id": "P-H6Lw1vyNNd"
},
"source": [
"\n",
"\n",
"## Machine Learning Examples\n",
"\n",
"To see end-to-end examples of the interactive machine learning analyses that Colaboratory makes possible, check out these tutorials using models from [TensorFlow Hub](https://tfhub.dev).\n",
"\n",
"A few featured examples:\n",
"\n",
"- [Retraining an Image Classifier](https://tensorflow.org/hub/tutorials/tf2_image_retraining): Build a Keras model on top of a pre-trained image classifier to distinguish flowers.\n",
"- [Text Classification](https://tensorflow.org/hub/tutorials/tf2_text_classification): Classify IMDB movie reviews as either *positive* or *negative*.\n",
"- [Style Transfer](https://tensorflow.org/hub/tutorials/tf2_arbitrary_image_stylization): Use deep learning to transfer style between images.\n",

```
            "- [Multilingual Universal Sentence Encoder Q&A](https://tensorflow.org/hub/tutorials/retrieval_with_tf_hub_universal_encoder_qa): Use a machine learning model to answer questions from the SQuAD dataset.\n",
            "- [Video Interpolation](https://tensorflow.org/hub/tutorials/tweening_conv3d): Predict what happened in a video between the first and the last frame.\n"
          ]
        }
    ]
}
```