## Overview

Often doctors use tools that estimate how likely a patient will develop an illness. It compiles a number of contributing factors, uses statistical data for each, and then computes the likelihood it will happen.

Naturally, this is not a guarantee. But it is a very helpful guide.

For this lab, you are going to create a Risk Estimator program that will ask a series of yes/no questions and, then, print a percentage to the screen.

Each question is worth a predetermined number of points. If the user specifies the correct answer, it is added to their total score.

From this, you can compute a Risk Estimate from the answers.
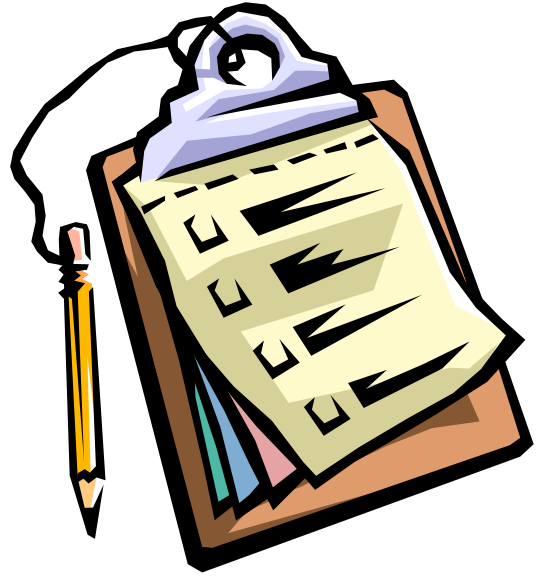
$$\textbf{Risk Estimate} \ = \frac{\textbf{Points Earned}}{\textbf{Total Points}}$$

Sample Run

The following is a sample run of the program. ***This is not a real medical test***. The user's input is printed in **blue**. The data outputted from your calculations is printed in **red**.

```
20 Year Heart Risk Estimator

Are you over 40 years old?          1 point
y

Are you a smoker?              Rest are 4 points
n

Do you have diabetes?
n

Do you have hypertension?
n

Do you drink?
n
                           1 / 17
Your risk estimate is 5%.
```

Have Fun!

Don't use medical questions above. Create your own risk estimator with your own questions and point values.

The following are some example categories.

- Risk of failing an exam
- Risk of barfing at a party
- Risk of getting into an auto accident
- Risk of getting "pwn'd" in a video game
- Risk of having explosive diarrhea
- etc...

Requirements

You <u>must</u> think of a solution on your own. The requirements are as follows:

1. **Prompt the user for each question**

2. **Use a <u>different</u> theme (and questions) than the example**.
   Labs using the medical questions above will not receive credit.

3. **Read a 'y' or 'n' answer for each question.**
   <u>No</u> credit will be given on the lab if y/n is not used.

4. **Print out the correct percentage.**
   How do you get a value greater than zero? That's your challenge to solve.

5. **At least 5 questions.**

Reading Text

To read text from the keyboard, please read about the ScanChar subroutine in the CSC35 Library.

Hints

- Like all labs, **build it in pieces**. First get a single If-Statement to work. Then, you can work on the more detailed ones.

- All labels **must** be unique. Choose your names well.

- Remember to use 8-bit registers to compare 'y' and 'n'.

Submitting Your Lab

Run Alpine by typing the following and, then, enter your username and password.

```
alpine
```

To submit your lab, send the source file (not a.out or the object file) to:

```
dcook@csus.edu
```

UNIX Commands

Editing

Action	Command	Notes
Edit File	**nano** *filename*	"Nano" is an easy to use text editor.
E-Mail	**alpine**	"Alpine" is text-based e-mail application. You will e-mail your assignments it.
Assemble File	**as −o** *objectfile asmfile*	Don't mix up the *objectfile* and *asmfile* fields. It will destroy your program!
Link File	**ld −o** *exefile objectfiles*	Link and create an executable file from one (or more) object files

Folder Navigation

Action	Command	Description
Change current folder	**cd** *foldername*	"Changes Directory"
Go to parent folder	**cd ..**	Think of it as the "back button".
Show current folder	**pwd**	Gives a file path
List files	**ls**	Lists the files in current directory.

File Organization

Action	Command	Description
Create folder	**mkdir** *foldername*	Folders are called directories in UNIX.
Copy file	**cp** *oldfile newfile*	Make a copy of an existing file
Move file	**mv** *filename foldername*	Moves a file to a destination folder
Rename file	**mv** *oldname newname*	Note: same command as "move".
Delete file	**rm** *filename*	Remove (delete) a file. There is **no** undo.