Ryan Bergquist

Choong Huh

CS460

Project1

1. **Where do *current\_thread* and *stored\_thread* reside in memory?**

They reside in the heap, assuming the memory has been allocated.

1. ***Stored\_thread’s* stack does not have to be allocated. Why not?**

The stack for stored\_thread is within the memory already allocated for current\_thread.

**3)** **Run your program with *current\_thread’s* stack size from 128, 256, 512, 1024, 2048 Bytes. What do you expect for the outputs?**

**What are the actual outputs and explain why?**

We should assume for sizes less then 512, due to the number of registers, there is not enough memory allocated.

Each register is set to be using 64 bits.

With memory below 512, there is not enough memory to hold enough for the registers. will cause a segmentation fault.

**4) Notice that if main finishes first, the other thread's function will not get to finish.**

**Why is this?**

When main thread terminates, all child processes also terminate. All registers and pointers are then freed leaving the computer in a state where it will no longer reach the child process.

**5) Notice that if the other thread's function finishes first, the program will crash with a Segementation Fault.**

**Why is this?**

When the thread finished, memory is cleaned. The main process tries to access memory in its function having an invalid pointer to memory. An invalid pointer leads to a seg-fault.