TIC1001—Introduction to Computing and Programming
National University of Singapore

Tutorial 09

Question 1

[Process Management] In the lecture, we discussed that OS is responsible for selecting a process to run on CPU. This is formally known as process scheduling. One way to make a good process scheduling decision is based on the process behavior. A process perform either:

- Input/Output (IO)-Activity: e.g. waiting for user input (scanf()), reading from a file (fgets()), printing information on screen (printf()), etc. OR
- CPU-Activity: e.g. performing heavy calculation.

(a) Discuss the behavior of the following processes:

1. The “desktop” interface commonly seen in most modern OSes.
2. The “Sieve of Erathoses” on a HUGE array (say 500 million entries).

In [ ]:
```
1. IO activity
2. CPU activity
```

(a)(i) Given the two processes in part (a), modern OS will attempt to prioritize which process?

In [ ]:
```
The "desktop" process
```

(b) On a system with an active user (e.g. you are coding actively on your laptop), discuss how should the OS prioritizes processes based on their behavior. As a guide, you should be thinking about, “if a process P is doing a lot I/O activity, then . . . ”. You can also use (a) as a guide.

Question 2

[File Management] Let us write a simple C program simulate how the OS uses File Allocation Table (FAT) to find out the disk blocks used by a file.

In [ ]:
```
#include <iostream>

using namespace std;

int readFAT( FILE* input, int fat[])
{
    int size, i;
    fscanf(input,"%d",&size);
    for (int i = 0; i < size; i++)
    {
        fscanf(input,"%d",&(fat[i]));
    }
    return size;
}

void printFAT( int fat[], int size)
{
    int i;
    for ( i = 0; i < size; i++)
    {
        printf("%3d[%3d]",i,fat[i]);
    }
}

void printFileAt( int fat[], int start)
{
    int idx = start;
    while (idx != -1)
    {
        printf("%d",idx);
        idx = fat[idx];
    }
    printf("\n");
}

int main(void)
{
    FILE* input;
    char filename[999];
    int i, fat[1024], size, start;

    printf("File name: ");
    scanf("%s",filename);

    input = fopen(filename,"r");
    size = readFAT(input,fat);
    printFAT(fat,size);

    while (scanf("%d",&start)==1)
    {
        printFileAt(fat,start);
    }
}
```

(2a) Select all TRUE statements regarding virtual memory management

In [ ]:
```
- It is possible to have memory pages of a process not reside in memory.
- More processes can now be residing in the memory.
```

Question 3

[Database] Given the following person table:

name	height	weight	age	IQ
Abigail	152.5	55	17	180
Trigun	175.4	85	25	135
Shelcard	195.0	120	55	120
Maurice	145.4	48	46	108
Liam	170.5	60	18	100

(a) Give the SQL statement to create the table above. You can make reasonable assumptions on the datatype of each column.

In [ ]:
```
CREATE
    TABLE
        person (
            name VARCHAR (256)
            ,height FLOAT
            ,weight INT
            ,age INT
            ,IQ INT
        )
```

(b) Give the SQL statements to insert the 5 rows above.

In [ ]:
```
INSERT
    INTO
        person (
            name
            ,height
            ,weight
            ,age
            ,IQ
        )
    VALUES (
        "Abigail"
        ,152.5
        ,55
        ,17
        ,180
    ) (
        "Trigun"
        ,175.4
        ,85
        ,25
        ,135
    ) (
        "Shelcard"
        ,195.0
        ,120
        ,55
        ,120
    ) (
        "Maurice"
        ,145.4
        ,48
        ,46
        ,108
    ) (
        "Liam"
        ,170.5
        ,60
        ,18
        ,100
    )
```

(c) Give the SQL statements to find out the following information. Note that you are free to make reasonable assumptions.

i. Young Genius

In [ ]:
```
SELECT
        *
    FROM
        person
    WHERE
        IQ > 120 AND age < 25
```

ii. Giant among us

In [ ]:
```
SELECT
        name
    FROM
        person
    WHERE
        height > 180 AND weight > 100
```

iii. The Joes (not tall, not bulky, not genius, i.e. average person)

In [ ]:
```
SELECT
        *
    FROM
        person
    WHERE
        height <= 180
        AND IQ <= 120
        AND weight <= 100
```

(d) Suppose we want to list all persons (name, height) in descending order of their heights, which of the following SQL statement is correct?

In [ ]:
```
SELECT
        name, height
    FROM
        person
    ORDER BY height DESC
```