

**Question 1** What is the difference between a C string and any `char` array?

- ☐ C string uses the last character to indicate the end of the string.
- ☐ `char` array does not store any quote characters
- ☐ C string is stored in static memory space
- ☐ no difference

**Question 2** Which of the following is the current capital of China?

- ☐ Shantou
- ☐ Shanghai
- ☐ Honk Kong
- ☐ Guangzhou
- ☐ Shenzhen
- ☐ *None of these answers are correct.*

**Question 3** ♣

Read the following code.

```
#define INITIAL_SEMAPHORE_VALUE 2
sem_t mysem;
int count = 0;

void *updatecount(void *arg) {
    sem_wait(&mysem);
    count++;
    sem_post(&mysem);
    return (void*)(0);
}
int main(int argc, char **argv) {
    sem_init( &mysem, ..., INITIAL_SEMAPHORE_VALUE );
    int i;
    for (i = 0; i < 4; ++i) {
        pthread_create(..., ..., updatecount, ...);
    }
    while (1); return 0;
}
```

What are the possible values of `count` after 4 threads have been created to execute the function `updatecount`.

- ☐ 0
- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

**Question 33 (Programming) (16 marks)**

This question considers a 2D grid of floating point numbers with dimensions **width** and **height**. Find the position of this grid that has the highest magnitude average. This function computes the sum of the neighbouring values in the 2D grid (NorthWest, North, NorthEast, East, SouthEast, South, SouthWest, West) plus the centre and divides this by 9. Neighbours outside the dimensions of the array are treated as value zero. The grid is given as a 1D array which stores the 2D grid positions row-major order.

The 1D array sequence: 1, 2, 3, 4, 5, 6, 7, 8, 9 represents the grid:

1	2	3
4	5	6
7	8	9

$$f(x, y) = \frac{1}{9} \sum_{\substack{x-1 < i < x+1 \\ y-1 < j < y+1}} |array(i, j)|$$

Write the code needed for the following function prototype to work. You may write additional helper functions if needed.

```
// pre: w > 0 h > 0
// returns the position of the highest magnitude average of the array
// in variables found_x and found_y
void get_hma(float *array, int w, int h, int *found_x, int *found_y);
```

**Parallel part**

Write a parallel solution using 4 threads to gain a speedup > 1. Write your solution with notation that is *similar* to pthreads. thread\_create(), thread\_join(), lock(mutex), unlock(mutex).

**Question 35 (Programming) (24 marks)**

Implement the functions required for the integer stack data structure in C.  
More explanation about a stack data structure...

```
// initialise any fields of struct stack
void create( struct stack * , int max_size );

// places the integer x on top of the stack
// when stack is full, undefined behaviour
void push( struct stack *, int x );

// returns the integer on top of the stack
// when stack is empty, undefined behaviour
int pop( struct stack * );

// returns non-zero value when empty
int isEmpty( );

// cleanup associated memory
void destroy( struct stack * );

struct stack
{
// define your struct here

};
```

Useful functions

```
void *malloc(size_t size);
void free(void *ptr);
```

Write some answer code...

**Parallel part**

Write the code needed to make the stack access safe by preventing race condition. Use *similar* notation to pthreads.