Pointers && Arrays

1. Write a program that takes two arrays, *speeds* and *durations* each with length n. If the index is an even number, then the robot should move forward with the given speed and duration and if the index is odd the robot should move backward with the given speed and duration

```
void
vibrate(double speeds[], double durations[], int n){
```



```
}
```

2. Suppose

```
int a[] = {5, 15, 34, 54, 14, 2, 52, 72};
int *p = &a[1];
int *q = &a[5];
```

   (a) What is the value of *(p+3)?
   (b) What is the value of *(q-3)?
   (c) Is the condition $p < q$ true of false?
   (d) Is the condition *p ¡ *q true of false?

3. Write a program $find\_largest$ that finds the largest element in array a of length n and store the value in the variable *largest is pointing to.

4. What will the contents of the array a be after the following code is executed?

```
#define N 10

int a[N] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
int *p = &a[0], *q = &a[N-1], temp;

while (p < q) {
    temp = *p;
    *p++ = *q;
    *q-- = temp;
}
```

$a =$

5. Write a program to sum the elements of array a with length n and return its average.

```
int *
compute_avg(int a[], int n){




}
```

Preprocessor

1. Write parameterized macros that compute the following

   (a) The cube of x.
   (b) The remainder when n is divided by 4.
   (c) 1 if the product of x and y is less than 100, 0 otherwise.

2. Let DOUBLE be the following macro:

$$\#define \ DOUBLE(X) \ 2*X$$

   (a) What is the value of DOUBLE(1+2)
   (b) What is the value of 4/DOUBLE(2)
   (c) Will this macro perform as expect? If not, fix it.

Stack Diagrams

1. Give the Stack Diagram and Program Counter for points A-D of the following program.

```
int                                     //1
compute(int  x,  int  y,  double  d)  { //2
    //  Point  A,  C——both  calls        //3
    return  (int)  (x * d + y * d);      //4
}                                        //5
                                        //6
int                                     //7
main(void)  {                           //8
    int  ret = 0;                        //9
    ret = compute(3,  5,  2.0);          //10
    //  Point  B                         //11
    compute(1,  10,  −1.0);              //12
    //  Point  D                         //13
    return  0;                           //14
}                                        //15
```

2. Give the Stack Diagram and Program Counter for points A-C of the following program.

```
int                                        //1
main(void) {                               //2
    int result = 1;                        //3
    // Point A                             //4
    for (int i = 4; i >= 0; i = i - 2) {   //5
        // Point B—first iteration only    //6
        result = result + i * 2 % 3;       //7
        printf("%d ", result);             //8
        // Point C—final iteration only    //9
    }                                      //10
    return 0;                              //11
}                                          //12
```

3. Give the Stack Diagram for points A-D of the following program.

```c
void
foo(int *px, int *py, int val) {  //1
    // Point B                    //2
    *px = *px + val;              //3
    *py = *py - val;              //4
    // Point C                    //5
}                                 //6
                                  //7
int                               //8
main(void){                       //9
    int x = 0;                    //10
    int y = 10;                   //11
    // Point A                    //12
    foo(&x, 5);                   //13
    // Point D                    //14
    return 0;                     //15
}                                 //16
```

Function Pointers

1. Write a function *map*, which takes an array, it's size, and a function that takes in an int and returns an int. *map* then updates the array such that the function is applied to every element.

```
void map(int arr[], int size, int func (int)){




}
```

2. What is the output of the following program

```
#include <stdio.h>

int f1 (int (*f) (int));
int f2(int i);

int main(void)
{
    printf("Answer: %d\n", f1(f2));
    return 0;
}

int f1(int (*f)(int))
{
    int n = 0;
    while ((*f)(n))
    {
        n++;
    }

    return n;
}

int f2(int i)
{
    return i * i + i - 12;
}
```

Binary Representation

1. What is the binary decimal representation of 71?

2. What is the signed magnitude representation of -71?

3. What is the ones complement representation of -71?

4. What is the twos complement representation of -71?

5. What is the IEEE floating point representation of -71?

Bitwise Operations

1. Determine integer that results from the following expressions

    (a) 7 & 5
    (b) 8 | 3
    (c) 9 << 2
    (d) 6 ∧ 3
    (e) ∼ (-3)

2. Suppose that s is the following structure

```
struct shape {
        int shape_kind;
        union
        {
                struct
                {
                    int height, width;
                } rectangle;
                struct
                {
                    int radius
                } circle;
        }u;
} s;
```

Indicate which of the following statements are legal, and show how to repair the ones that aren't

    (a) s.shape_kind = 1;
    (b) s.height=25;
    (c) s.u.rectangle.width =8;
    (d) s.u.circle =5;