

TEST 3, CS 101, FRIDAY, APRIL 16, 2004

(1) What is the output?

```
#include<iostream.h>
void print(int[][3]);
void print_warp(int[][3]);
void main(){
    int a[2][3], i, j;
    for (i=0; i<2; i++)
        for (j=0; j<3; j++)
            a[i][j] = i + 2*j + 1;
    print(a);
    cout << endl;
    print_warp(a);
}
void print(int b[][3]){
    for (int i=0; i<2; i++){
        for (int j=0; j<3; j++)
            cout << b[i][j] << " ";
        cout << endl;
    }
}
void print_warp(int b[][3]){
    for (int j=0; j<3; j++){
        for (int i=0; i<2; i++)
            cout << b[i][j] << " ";
        cout << endl;
    }
}
```

Output:

**1 3 5**  
**2 4 6**

**1 2**  
**3 4**  
**5 6**

(2) What is the output of the following program?

```
#include <iostream.h>
void main(){
    int a[5]={1,6,4,3,2};
    for (int i=0; i<4; i++) cout << a[i+1]+1 << endl;
    for (int i=1; i<5; i++) a[i] += a[i-1];
    for (int i=1; i<5; i++) cout << a[i-1] << endl;
}
```

Output:

```
7
5
4
3
1
7
11
14
```

(3) What is the output?

```
#include<iostream.h>
void main(){
    int a[]={4,6,9,-2,3,5}, *b;
    b=a;
    for (int i=0; i<3; i++){
        cout << *b << endl;
        b+=2;
    }
}
```

Output:

```
4
9
3
```

- (4) Use exchange sort to sort the following array. Indicate all the swaps that are made, and write out the array following each swap.

```
10 7 1 4 0 3
 3 7 1 4 0 10 (swap 10 and 3)
 3 0 1 4 7 10 (swap 7 and 0)
 3 0 1 4 7 10 (no swap)
 1 0 3 4 7 10 (swap 3 and 1)
 0 1 3 4 7 10 (swap 1 and 0)
```

- (5) For each integer  $n$ ,  $a_n$  is defined to be 1 if  $n \leq 1$ , and is defined by the formula  $a_n = na_{n-2}$  for  $n > 1$ . Write a function

```
int a(int n)
whose output is  $a_n$ . To implement this function, YOU MUST
USE A RECURSIVE FUNCTION CALL.
```

```
int a(int n){
    if (n<=1)
        return 1;
    else
        return n*a(n-2);
}
```

- (6) Write a function  
`int min(int x[], int n)`  
that returns the minimum of `x[0], x[1], ..., x[n-1]`.

```
int min(int x[], int n){
    int i, m=x[0];
    for (i=1; i<n; i++)
        if (x[i]<m)
            m = x[i];
    return m;
}
```

- (7) Write a function  
`int pass(double gr[], int n)`  
that returns the number of values of `gr[0], gr[1], ..., gr[n-1]`  
which are  $\geq 60$ . (If `gr` represents a list of student grades, then  
this function will count the number of passing grades.)

```
int pass(double gr[], int n){
    int p=0, i;
    for (i=0; i<n; i++)
        if (gr[i]>=60)
            p++;
    return p;
}
```

- (8) Consider the following class header file `myclass.h`:

```
#include<iostream.h>
class myclass{
private: int x, y;
public: void print();
        void assign(int,int);
        myclass swap();
};
// Implementation
void myclass::assign(int m, int n)
    {x=m; y=n;}
myclass myclass::swap()
    {myclass s; s.x=y; s.y=x;
    return s;}
void myclass::print()
    {cout << x << " " << y << endl;}
```

- (a) List the function members of the class `myclass`.

```
print
assign
swap
```

- (b) This class is used by the following program. Give the output of the program in the space provided:

```
#include "myclass.h"
void main(){
    myclass p, q, r;
    p.assign(4,0); q.assign(2,4);
    r.assign(1,2);
    p.print(); q.print();
    q = r.swap();
    q.print(); r.print();
}
```

Output:

```
4 0
2 4
2 1
1 2
```