

Programming Assignment 12
Review for test

YOU DO NOT HAVE TO HAND THIS IN

Print Your Name and Student #: _____

There are two parts to this assignment

Part A

To help you review for the test, answer questions 3, 5, 6, 7 and 8 in section 5.8 of the text book. They are attached below. Thus, print out this sheet and each person must hand in a solution.

- 3) Show the screen display for the following C program.

```
#include<stdio.h>

int main()
{ // begin main
  int k, new_ans, i=0;
  int v[ ] = {-1,0,1,2,3,4,5};

  for ( k=5; k>0; k=k-1 )
  {
    switch (i)
    {
      case 0 :
      case 2 : new_ans = v[k]+v[k+1];
               printf( "k = %d\n", k );
               break;
      case 1 : new_ans=k;
               printf( "i = %d\n", i );
               printf( "k = %d\n", k );
               break;
      case 3 : new_ans = v[k]-v[k+1];
               printf( "i = %d\n", i );
               break;
      default: new_ans=i;
    }
    printf( "new_ans = %d\n", new_ans );
    i=i+1;
  }
} // end main
```

Display #	Display
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	

- 5) What will be displayed by the following program? Pay attention to formatting in the **printf** statement.

```
#include<stdio.h>
int main()
{ // begin main

    int a = 5, b = 3, c = 4, d = 2, e, f;
    float alpha = 3.0, beta = 2.0, gamma = 4.0, delta, epsilon;

    delta = beta*a/d;
    e = a/c;
    epsilon = (gamma/alpha)*(a/d);
    beta = (2.0/3.0)*beta;
    f = b%d;

    printf( "\ndelta = %.3f \ne = %2d \nepsilon = %6.2f"
           "\nbeta = %f \nf = %d \n", delta, e, epsilon, beta, f);

} // end main
```

delta =
 e =
 epsilon =
 beta =
 f =

- 7) Rewrite the following program so that it uses a do-while loop in place of the for loop.

```
#include <stdio.h>
int main()
{
    int i, n = 4;
    float x = 1.0;
    for (i = 1; i <= n; i++) {
        x = 1 / x;
        x = x + 1;
        printf("%d %.4f\n", i, x);
    }
}
```

6) Show the screen display for the following C program.

```
#include<stdio.h>

int main()
{ // begin main
    int k, kk, a, b, c=0;
    int v[ ]={5, 7, 8, 10, 17, 4};

    for ( k=1 ; k<=11; k=k+1 )
    {
        c=c+1;
        if ( k>3 && k<=5 )
        {
            kk = v[k];
            printf("\nfirst branch");
            printf( "\nkk = %d", kk );
        }
        else if ( k<=8 && k>3 && k!=7)
        {
            b = k*2;
            printf("\nsecond branch");
            printf( "\nb = %d", b );
        }
        else if ( k==10 || k==2 )
        {
            a = k;
            printf("\nthird branch");
            printf( "\na = %d", a );
        }
        else if ( k>=8 )
        {
            printf("\nlast branch");
            printf( "\nc = %d", c );
        }
    }
    printf( "\nDone\n\n" );
} // end main
```

Display #	Display
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	

8) The following program is one solution to your programming assignment 12. It is written with function calls from main(). The function calls (underlined and in bold) are as shown. Referring to both the function calls and the function definitions (provided without prototype), write appropriate function prototypes and definitions in the space provided.

```

#include<stdio.h>
#include<math.h>
#define MAXSIZE 50
// PROTOTYPES:

_____// enter prototype for header

_____// enter prototype for getxydata

_____// enter prototype for getangle

_____// enter prototype for interp

_____// enter prototype for displayresults

_____// enter prototype for goagain

//*****
int main()
{ // begin main
  // variable declarations
    double lift[MAXSIZE], angle[MAXSIZE];
    int numpts;
    double liftangle, liftest;
  // algorithm
    header();
    // get lift data
    getxydata( angle, lift, &numpts );
    // estimate lift given angle
    do
    { // begin do..while
      liftangle = getangle( );
      liftest = interp( liftangle, angle, lift, numpts );
      displayresults( liftangle, liftest );
    } while ( goagain() == 1 );

} // end main
//*****

_____header(_____ ) // complete fcn definition
{ // begin header
  printf( "\nThis is an exam question!!!!\n\n" );
} // end header

_____displayresults(_____ ) // complete fcn definition
{ // begin displayresults
  printf( "\n\nLift angle = %.3lf", liftangle );
  printf( "\n\nEstimated lift = %.3lf", liftest );
} // end displayresults
//

```

```

_____getxydata(_____ ) // complete fcn definition
{ // begin getxydata
// variable declaration
FILE *pInfile;
// algorithm
pInfile = fopen("liftdata.dat", "r");;
*pnumpts = 0;
while ( fscanf(pInfile, " %lf %lf",&x[*pnumpts],&y[*pnumpts] ) == 2 ){
    *pnumpts = *pnumpts+1; } // end while
fclose(pInfile);
} // end getxydata

```

```

_____getangle(_____ ) // complete fcn definition
{ // begin getangle
// variable declaration
double angle;
// algorithm
printf( "\n\nPlease enter angle for lift estimate ==> " );
scanf( "%lf", &angle );

return( angle );
} // end getangle

```

```

_____interp(_____ ) // complete fcn definition
{ // begin interp
// variable declaration
double y, slope;
int i;
// algorithm
if ( x <= xvec[0] ) {
    y = yvec[0]; }
else if ( x >= xvec[npts-1] ) {
    y = yvec[npts-1];}
else {
    i = 0;
    while ( x > xvec[i] ) {
        i = i+1; }
    slope = (yvec[i]-yvec[i-1])/(xvec[i]-xvec[i-1]);
    y = yvec[i]+slope*(x-xvec[i]); }

return(y);
} // end interp

```

```

_____goagain( _____ ) // complete fcn definition
{ // begin goagain
// variable declaration
int choice;
// algorithm
printf( "\n\nAnother estimate? 1 = yes, 2 = no ==> " );
scanf( "%d", &choice );
return( choice );
} // end goagain } // end goagain

```

Part B

Find the 7 errors in the following program:

1)

```
#include<stdio.h>

int main()
{ // begin main
  int k, new_ans, i=0;
  int v( ) = {-1,0,1,2,3,4,5};

  for ( k=5, k>0, k=k-1 )
  {
    switch i
    {
      case 0 :
        case 2 new_ans = v[k]+v[k+1];
        printf( "k = %d\n", k );
          break;
      case 1 : new_ans=k;
        printf( "i = %d\n", i );
        printf( "k = %f\n", k );          break;
      case 3 : new_ans = v[k]-v[k+1];
        printf( "i = %d\n", i );
          break;
      otherwise: new_ans=i;          }
    printf( "new_ans = %d\n", new_ans );
    i=i+1;
  } // end main
```

Find the error and fix it: _____

Find the error and fix it: _____

Find the error and fix it: _____

Find the error and fix it: _____

Find the error and fix it: _____

Find the error and fix it: _____

Find the error and fix it: _____