Instructions:
1. Print your name in the space provided
2. Print your student identifier in the space provided.
3. Print the section number of the section in which you are enrolled in the space provided.
4. Print the date in the space provided.
5. You have 50 minutes to complete this examination.
6. You may use a calculator.
7. Reference materials are prohibited. You must complete this test without the aid of course notes or reference texts.
8. Questions requiring written answers must be answered using standard American English. Answers containing spelling or grammatical errors will be given no credit.
9. Answers must be coded legibly. Answers that cannot be read by your instructor will be given no credit.
10. You must do your own work.

Scoring:
1. The table to the right lists the number of raw points available for each problem and the total number of raw points that can be earned on this test.
2. Your score will be normalized to a fraction of 150 points. If $n$ is your normalized score, $r$ is your raw score, and $T$ is the total number of raw points, then $n = 150 \frac{r}{T}$.
1. (54 points) Code a C program that prints THE CANARY by Ogden Nash as shown below.

THE CANARY

The song of canaries
Never varies,
And when they're moulting
They're pretty revolting.

Ogden Nash

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>#include &lt;stdio.h&gt;</code></td>
</tr>
<tr>
<td>2</td>
<td><code>int main()</code></td>
</tr>
<tr>
<td>3</td>
<td><code>{ printf(&quot;THE CANARY\n&quot;);</code></td>
</tr>
<tr>
<td>4</td>
<td><code>printf(&quot;\n&quot;);</code></td>
</tr>
<tr>
<td>5</td>
<td><code>printf(&quot;The song of canaries\n&quot;);</code></td>
</tr>
<tr>
<td>6</td>
<td><code>printf(&quot;Never varies,\n&quot;);</code></td>
</tr>
<tr>
<td>7</td>
<td><code>printf(&quot;And when they're moulting\n&quot;);</code></td>
</tr>
<tr>
<td>8</td>
<td><code>printf(&quot;They're pretty revolting.\n&quot;);</code></td>
</tr>
<tr>
<td>9</td>
<td><code>printf(&quot;\n&quot;);</code></td>
</tr>
<tr>
<td>10</td>
<td><code>printf(&quot;Ogden Nash\n&quot;);</code></td>
</tr>
<tr>
<td>11</td>
<td><code>return 0;</code></td>
</tr>
<tr>
<td>12</td>
<td><code>}</code></td>
</tr>
</tbody>
</table>

Total: 54

2. (25 points) What does the program given below print?

```c
#include <stdio.h>
int main()
{ int a=73,b=13;
  printf("%s
%s
%s
%s
%s
","a =",a
,"b =",b
,"a / b =",a/b
,"a % b =",a%b
,"ANSI check =",(a/b)*b+a%b-a
);
  return 0;
}
```

```
  a = 73
  b = 13
  a / b = 5
  a % b = 8
  ANSI check = 0
```
3. (53 points) Code a C program that will produce the dialog given below. Bold-faced values are entered by the user. Function irate issues the prompt and reads the annual percentage rate. Function irate returns the monthly interest rate. Function main prints the second line i=0.006875. Show the dialog printed by the program listed given 8.25 was entered on the command line after the prompt.

```c
#include <stdio.h>
double irate(void)
{ double APR;
    printf("Enter the annual percentage rate earned on your account. ");
    scanf("%lf",&APR);
    return APR/1200;
}
int main()
{ printf("i=%.6f\n",irate());
    return 0;
}
```

Enter the annual percentage rate earned on your account. 8.25
i=0.006875

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>#include &lt;stdio.h&gt;</code></td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td><code>double irate(void)</code></td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td><code>{ double APR;</code></td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td><code>      printf(&quot;Enter the annual percentage rate earned on&quot;);</code></td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td><code>      printf(&quot; your account. &quot;);</code></td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td><code>      scanf(&quot;%lf&quot;,&amp;APR);</code></td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td><code>      return APR/1200;</code></td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td><code>}</code></td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td><code>int main()</code></td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td><code>{ printf(&quot;i=%.6f\n&quot;,irate());</code></td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td><code>      return 0;</code></td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td><code>}</code></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>53</strong></td>
</tr>
</tbody>
</table>

4. (10 points) What is printed by the program you coded in problem 3 if the annual percentage rate entered is 7.5?

Enter the annual percentage rate earned on your account. 7.5
i=0.006250
5. (20 points) Derive a closed-form expression for the sequence given by \( S \) below.

\[
S = a + ar + ar^2 + \cdots + ar^n
\]

\[
rS = ar + r^2 + \cdots + ar^n + ar^{n+1}
\]

\[
S = a + ar + ar^2 + \cdots + ar^n
\]

\[
rS - S = -a + ar^{n+1}
\]

\[
S = a \frac{r^{n+1} - 1}{r - 1}
\]

6. (20 points) An amount \( R \) is deposited every month for \( n \) months in an interest bearing account. Interest is compounded monthly. The periodic interest rate is \( i \). The future value of a deposit is \( R(1+i)^k \) where that deposit accumulated interest for \( k \) months. Find the sequence, \( S \), that represents the future value for all deposits.

\[
S = R + R(1+i) + R(1+i)^2 + \cdots + R(1+i)^n
\]

7. (20 points) Find an expression, \( S \), for the future value of the current bank balance, \( P \), and a sequence of equal monthly deposits, \( R \), into an account having a periodic interest rate, \( i \), that is compounded monthly.

\[
S = P(1+i)^n + R \frac{(1+i)^{n+1} - 1}{i}
\]

8. (10 points) How much money will you have in your account after making monthly deposits of $100 for 30 years on an account that yields 12 APR, compounded monthly. The initial balance is $1000.

$389,041.02
9. (94 points) Code a C program that will compute the value you found in problem 8. Code function \( f_v \) that returns the future value of the account. Function \( f_v \) accepts the initial bank balance, \( P \), the amount of the monthly deposit, \( R \), the periodic interest rate, \( i \), and the number of monthly deposits \( n \).

Function main prints "After 30 years you will have $x.y" where \( x \) is integer portion of the future value and \( y \) is the fractional portion of the future value.

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#include &lt;stdio.h&gt;</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>#include &lt;math.h&gt;</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>double ( f_v )(double ( P ), double ( R ), double ( i ), int ( n ))</td>
<td>15</td>
</tr>
<tr>
<td>4</td>
<td>{ return ( P ) * pow(( i+i ), ( n )) + ( R ) * (pow(( i+i ), ( n+1 )) - 1))/( i );</td>
<td>32</td>
</tr>
<tr>
<td>5</td>
<td>}</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>int main()</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>{ printf(&quot;After 30 years you will have $%.2f\n&quot; , ( f_v )(1000,100,12.0/1200,30*12)</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>);</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>return 0;</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>}</td>
<td>1</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>94</strong></td>
</tr>
</tbody>
</table>

10. (7 points) Write a command accepted by the campus computer that will compile and link the program you wrote in answer to problem 9. Assume the program is in file \( fv.c \) and the executable file is \( fv \).

\$ xlc -o fv fv.c -lm

11. (2 points) Write a command that will create file \( sea.c \) using one of the editors available on the campus computer.

\$ vi sea.c

12. (2 points) Write a command accepted by the campus computer that will print file \( sea.c \).

\$ lp sea.c
13. (20 points) What does the program given below print?

```
#include <stdio.h>
int main()
{
    int a=4,b=3,c=2,d=1;
    printf("%d\n",a*b/c%a);
    a=4; b=3; c=2; d=1;
    printf("%d\n",a*b%c+1);
    a=4; b=3; c=2; d=1;
    printf("%d\n",++a*b-c--);
    a=4; b=3; c=2; d=1;
    printf("%d\n",--d+c*-a/b++);
    return 0;
}
```

2
1
13
-2

14. (25 points) What does the program given below print?

```
#include <stdio.h>
int main()
{
    int i,j,k,m;
    i=4; j=3; k=2; m=1;
    printf("%d\n",i+=j+k);
    i=4; j=3; k=2; m=1;
    printf("%d\n",j*=k+=m=5);
    i=4; j=3; k=2; m=1;
    printf("%d\n",k*=m+=2);
    i=4; j=3; k=2; m=1;
    printf("%d\n",m*++i%j--);
    i=4; j=3; k=2; m=1;
    printf("%d\n",m/=(i+=k+=3+m));
    return 0;
}
```

9
21
6
2
0
15. (25 points) What does the program given below print?

```c
#include <stdio.h>
int main()
{
    int i, j, k;
    double x, y;
    i=10; j=2; k=6; x=3.2; y=-4.6;
    printf("%d\n", i==j);
    i=10; j=2; k=6; x=3.2; y=-4.6;
    printf("%d\n", !x==i);
    i=10; j=2; k=6; x=3.2; y=-4.6;
    printf("%d\n", i+j+k==2-k);
    i=10; j=2; k=6; x=3.2; y=-4.6;
    printf("%d\n", i!=j!=k&&x<y);
    i=10; j=2; k=6; x=3.2; y=-4.6;
    printf("%d\n", (x!=k) || (y=0));
    return 0;
}
```

16. (51 points) Write a computer program that will print powers of 3 from $3^0$ to $3^5$. You are prohibited from using standard library function pow in your solution.

```
#include <stdio.h>
int main()
{
    int e, p3=1;
    for (e=0; e<6; e++) {
        printf("%d ");
        p3*=3;
    }
    printf("\n");
    return 0;
}
```

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#include &lt;stdio.h&gt;</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>int main()</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>{ int e, p3=1;</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>for (e=0; e&lt;6; e++)</td>
<td>14</td>
</tr>
<tr>
<td>5</td>
<td>printf(&quot;%d &quot;);</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>p3*=3;</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>}</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>printf(&quot;\n&quot;);</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>return 0;</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>}</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>51</strong></td>
</tr>
</tbody>
</table>
17. (47 points) Rewrite the following program and replace the for-loop with equivalent assignment statements and while-statements.

```c
#include <stdio.h>
int main()
{
    int i,sum;
    for (i=1,sum=0;i<=10;sum+=i,i++);
    printf("sum=%d\n",sum);
    return 0;
}
```

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#include &lt;stdio.h&gt;</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>int main()</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>{ int i=1,sum=0;</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>while (i&lt;=10) {</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>sum+=i;</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>i++;</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>}</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>printf(&quot;sum=%d\n&quot;,sum);</td>
<td>9</td>
</tr>
<tr>
<td>9</td>
<td>return 0;</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>}</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>47</td>
</tr>
</tbody>
</table>
18. (45 points) What does the program given below print?

```c
#include <stdio.h>
int denom(int v, int d, char c)
{ while (v>=d) {
    v=v-d;
    printf("%c",c);
}
    return v;
}
void roman(int d)
{ d=denom(d,1500,'h');
  d=denom(d, 750,'g');
  d=denom(d, 125,'f');
  d=denom(d,  75,'e');
  d=denom(d,  25,'d');
  d=denom(d,   5,'b');
  d=denom(d,   1,'a');
}
int main()
{  int d=4597;
    printf("%6d ");
    roman(d);
    printf("\n");
    return 0;
}
```

4597 hhheccaa