

adder.c

1/1

lectures/1/src/

```
1: /*****
2:  * adder.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Adds two numbers.
8:  *
9:  * Demonstrates use of CS 50's library.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // ask user for input
19:     printf("Give me an integer: ");
20:     int x = GetInt();
21:     printf("Give me another integer: ");
22:     int y = GetInt();
23:
24:     // do the math
25:     printf("The sum of %d and %d is %d!\n", x, y, x + y);
26: }
```

conditions1.c

1/1

lectures/1/src/

```
1: /*****
2:  * conditions1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Tells user if his or her input is positive or negative (somewhat
8:  * innacurately).
9:  *
10: * Demonstrates use of if-else construct.
11: *****/
12:
13: #include <cs50.h>
14: #include <stdio.h>
15:
16: int
17: main(int argc, char *argv[])
18: {
19:     // ask user for an integer
20:     printf("I'd like an integer please: ");
21:     int n = GetInt();
22:
23:     // analyze user's input (somewhat inaccurately)
24:     if (n > 0)
25:         printf("You picked a positive number!\n");
26:     else
27:         printf("You picked a negative number!\n");
28: }
```

conditions2.c

1/1

lectures/1/src/

```
1: /*****
2:  * conditions2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Tells user if his or her input is positive or negative.
8:  *
9:  * Demonstrates use of if-else if-else construct.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // ask user for an integer
19:     printf("I'd like an integer please: ");
20:     int n = GetInt();
21:
22:     // analyze user's input
23:     if (n > 0)
24:         printf("You picked a positive number!\n");
25:     else if (n == 0)
26:         printf("You picked zero!\n");
27:     else
28:         printf("You picked a negative number!\n");
29: }
```

endowment.c

1/1

lectures/1/src/

```
1: /*****
2:  * endowment.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Simulates Harvard's endowment.
8:  *
9:  * Demonstrates \r, fflush, and sleep.
10: *****/
11:
12: #include <stdio.h>
13: #include <unistd.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // simulate progress from 0% to 100%
19:     for (long long l = 36900000000; l >= 26000000000; l--)
20:     {
21:         printf("\rEndowment: $%lld", l);
22:         fflush(stdout);
23:     }
24:     printf("\n");
25: }
```

```
1: /*****
2:  * f2c.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Converts Fahrenheit to Celsius.
8:  *
9:  * Demonstrates arithmetic.
10: *****/
11:
12:
13: #include <cs50.h>
14: #include <stdio.h>
15:
16: int
17: main(int argc, char *argv[])
18: {
19:     // ask user user for temperature in Fahrenheit
20:     printf("Temperature in F: ");
21:     float f = GetFloat();
22:
23:     // convert F to C
24:     float c = 5 / 9.0 * (f - 32);
25:
26:     // display result
27:     printf("%.1f F = %.1f C\n", f, c);
28: }
```

```
1: /*****
2:  * hai1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Says hello to the world.
8:  *
9:  * Demonstrates use of printf.
10: *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char *argv[])
16: {
17:     printf("O hai, world!\n");
18: }
```

hai2.c

lectures/1/src/

1/1

```
1: /*****
2:  * hai2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Says hello to just David.
8:  *
9:  * Demonstrates use of CS 50's library.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     string name = "David";
19:     printf("O hai, %s!\n", name);
20: }
```

hai3.c

lectures/1/src/

1/1

```
1: /*****
2:  * hai3.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Says hello to whomever.
8:  *
9:  * Demonstrates use of CS 50's library and standard input.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     printf("State your name: ");
19:     string name = GetString();
20:     printf("O hai, %s!\n", name);
21: }
```

```
1: // http://www.ioccc.org/years.html
2:
3: #include "stdio.h"
4: #define e 3
5: #define g (e/e)
6: #define h ((g+e)/2)
7: #define f (e-g-h)
8: #define j (e*e-g)
9: #define k (j-h)
10: #define l(x) tab2[x]/h
11: #define m(n,a) ((n&(a))==a)
12:
13: long tab1[]={ 989L,5L,26L,0L,88319L,123L,0L,9367L };
14: int tab2[]={ 4,6,10,14,22,26,34,38,46,58,62,74,82,86 };
15:
16: main(m1,s) char *s; {
17:     int a,b,c,d,o[k],n=(int)s;
18:     if(m1==1){ char b[2*j+f-g]; main(1(h+e)+h+e,b); printf(b); }
19:     else switch(m1-h){
20:     case f:
21:         a=(b=(c=(d-g)<<g)<<g)<<g);
22:         return(m(n,a|c)|m(n,b)|m(n,a|d)|m(n,c|d));
23:     case h:
24:         for(a=f;a<j;++a)if(tab1[a]&&!(tab1[a]%((long)l(n))))return(a);
25:     case g:
26:         if(n<h)return(g);
27:         if(n<j){n-=g;c='D';o[f]=h;o[g]=f;}
28:         else{c='r'-'\b';n-=j-g;o[f]=o[g]=g;}
29:         if((b=n)>=e)for(b=g<<g;b<n;++b)o[b]=o[b-h]+o[b-g]+c;
30:         return(o[b-g]%n+k-h);
31:     default:
32:         if(m1==e)main(m1-g+e+h,s+g); else *(s+g)=f;
33:         for(*s=a=f;a<e;) *s=(s<e)|main(h+a++,(char *)m1);
34:     }
35: }
36:
37:
```

```
1: /*****
2:  * math1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Computes a total but does nothing with it.
8:  *
9:  * Demonstrates use of variables.
10: *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char *argv[])
16: {
17:     int x = 1;
18:     int y = 2;
19:     int z = x + y;
20: }
```

```
1: /*****  
2:  * math2.c  
3:  *  
4:  * Computer Science 50  
5:  * David J. Malan  
6:  *  
7:  * Computes and prints an integral total.  
8:  *  
9:  * Demonstrates use of a format string.  
10: *****/  
11:  
12: #include <stdio.h>  
13:  
14: int  
15: main(int argc, char *argv[])  
16: {  
17:     int x = 1;  
18:     int y = 2;  
19:     int z = x + y;  
20:     printf("%d", z);  
21: }
```

```
1: /*****  
2:  * math3.c  
3:  *  
4:  * Computer Science 50  
5:  * David J. Malan  
6:  *  
7:  * Computes and prints a floating-point total.  
8:  *  
9:  * Demonstrates loss of precision.  
10: *****/  
11:  
12: #include <stdio.h>  
13:  
14: int  
15: main(int argc, char *argv[])  
16: {  
17:     float answer = 17 / 13;  
18:     printf("%.2f\n", answer);  
19: }
```

```
1: /*****  
2:  * math4.c  
3:  *  
4:  * Computer Science 50  
5:  * David J. Malan  
6:  *  
7:  * Computes and prints a floating-point total.  
8:  *  
9:  * Demonstrates use of floating-point math.  
10: *****/  
11:  
12: #include <stdio.h>  
13:  
14: int  
15: main(int argc, char *argv[])  
16: {  
17:     float answer = 17 / 13.0;  
18:     printf("%.2f\n", answer);  
19: }
```

```
1: /*****  
2:  * math5.c  
3:  *  
4:  * Computer Science 50  
5:  * David J. Malan  
6:  *  
7:  * Computes and prints a floating-point total.  
8:  *  
9:  * Demonstrates use of casting.  
10: *****/  
11:  
12: #include <stdio.h>  
13:  
14: int  
15: main(int argc, char *argv[])  
16: {  
17:     float answer = 17 / (float) 13;  
18:     printf("%.2f\n", answer);  
19: }
```

```
1: /*****
2:  * nonswitch.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Assesses the size of user's input.
8:  *
9:  * Demonstrates use of Boolean ANDing.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // ask user for an integer
19:     printf("Give me an integer between 1 and 10: ");
20:     int n = GetInt();
21:
22:     // judge user's input
23:     if (n >= 1 && n <= 3)
24:         printf("You picked a small number.\n");
25:     else if (n >= 4 && n <= 6)
26:         printf("You picked a medium number.\n");
27:     else if (n >= 7 && n <= 10)
28:         printf("You picked a big number.\n");
29:     else
30:         printf("You picked an invalid number.\n");
31: }
```

```
1: /*****
2:  * positives.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Demands that user provide a positive number.
8:  *
9:  * Demonstrates use of do-while.
10: *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char *argv[])
16: {
17:     int n;
18:
19:     do
20:     {
21:         printf("I demand that you give me a positive integer: ");
22:         n = GetInt();
23:     }
24:     while (n < 1);
25:     printf("Thanks for the %d.\n", n);
26: }
```



```
1: /*****
2:  * positive1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Demands that user provide a positive number.
8:  *
9:  * Demonstrates use of do-while.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     int n;
19:
20:     // loop until user provides a positive integer
21:     do
22:     {
23:         printf("I demand that you give me a positive integer: ");
24:         n = GetInt();
25:     }
26:     while (n < 1);
27:     printf("Thanks for the %d!\n", n);
28: }
```

```
1: /*****
2:  * positive2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Demands that user provide a positive number.
8:  *
9:  * Demonstrates use of bool.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     bool thankful = false;
19:
20:     // loop until user provides a positive integer
21:     do
22:     {
23:         printf("I demand that you give me a positive integer: ");
24:         if (GetInt() > 0)
25:             thankful = true;
26:     }
27:     while (thankful == false);
28:     printf("Thanks for the positive integer!\n");
29: }
```

```
1: /*****
2:  * positive3.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Demands that user provide a positive number.
8:  *
9:  * Demonstrates use of !.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     bool thankful = false;
19:
20:     // loop until user provides a positive integer
21:     do
22:     {
23:         printf("I demand that you give me a positive integer: ");
24:         if (GetInt() > 0)
25:             thankful = true;
26:     }
27:     while (!thankful);
28:     printf("Thanks for the positive integer!\n");
29: }
```

```
1: /*****
2:  * progress1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Simulates a progress bar.
8:  *
9:  * Demonstrates sleep.
10: *****/
11:
12: #include <stdio.h>
13: #include <unistd.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // simulate progress from 0% to 100%
19:     for (int i = 0; i <= 100; i++)
20:     {
21:         printf("Percent complete: %d%\n", i);
22:         sleep(1);
23:     }
24:     printf("\n");
25: }
```

```
1: /*****
2:  * progress2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Simulates a better progress bar.
8:  *
9:  * Demonstrates \r, fflush, and sleep.
10: *****/
11:
12: #include <stdio.h>
13: #include <unistd.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // simulate progress from 0% to 100%
19:     for (int i = 0; i <= 100; i++)
20:     {
21:         printf("\rPercent complete: %d%%", i);
22:         fflush(stdout);
23:         sleep(1);
24:     }
25:     printf("\n");
26: }
```

```
1: /*****
2:  * progress3.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Simulates a better progress bar.
8:  *
9:  * Demonstrates a while loop.
10: *****/
11:
12: #include <stdio.h>
13: #include <unistd.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     int i = 0;
19:
20:     /* simulate progress from 0% to 100% */
21:     while (i <= 100)
22:     {
23:         printf("\rPercent complete: %d%%", i);
24:         fflush(stdout);
25:         sleep(1);
26:         i++;
27:     }
28:     printf("\n");
29: }
```

```
1: /*****
2:  * sizeof.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Reports the sizes of C's data types.
8:  *
9:  * Demonstrates use of sizeof.
10: *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char *argv[])
16: {
17:     // some sample variables
18:     char c;
19:     double d;
20:     float f;
21:     int i;
22:
23:     // report the sizes of variables' types
24:     printf("char: %d\n", sizeof(c));
25:     printf("double: %d\n", sizeof(d));
26:     printf("float: %d\n", sizeof(f));
27:     printf("int: %d\n", sizeof(i));
28: }
```

```
1: /*****
2:  * switch1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Assesses the size of user's input.
8:  *
9:  * Demonstrates use of a switch.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // ask user for an integer
19:     printf("Give me an integer between 1 and 10: ");
20:     int n = GetInt();
21:
22:     // judge user's input
23:     switch (n)
24:     {
25:         case 1:
26:         case 2:
27:         case 3:
28:             printf("You picked a small number.\n");
29:             break;
30:
31:         case 4:
32:         case 5:
33:         case 6:
34:             printf("You picked a medium number.\n");
35:             break;
36:
37:         case 7:
38:         case 8:
39:         case 9:
40:         case 10:
41:             printf("You picked a big number.\n");
42:             break;
43:
44:         default:
45:             printf("You picked an invalid number.\n");
46:     }
47: }
```

```
1: /*****
2:  * switch2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Assesses a user's grade.
8:  *
9:  * Demonstrates use of a switch.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char *argv[])
17: {
18:     // ask user for a char
19:     printf("Pick a letter grade: ");
20:     char c = GetChar();
21:
22:     // judge user's input
23:     switch (c)
24:     {
25:         case 'A':
26:         case 'a':
27:             printf("You picked an excellent grade.\n");
28:             break;
29:
30:         case 'B':
31:         case 'b':
32:             printf("You picked a good grade.\n");
33:             break;
34:
35:         case 'C':
36:         case 'c':
37:             printf("You picked a fair grade.\n");
38:             break;
39:
40:         case 'D':
41:         case 'd':
42:             printf("You picked a poor grade.\n");
43:             break;
44:
45:         case 'E':
46:         case 'e':
47:             printf("You picked a failing grade.\n");
48:             break;
49:
50:         default:
51:             printf("You picked an invalid grade.\n");
52:     }
53: }
```