# F. 15 Chapter 15 Solutions

15.1

1. #include <stdio.h>

int main()

{

int i = 1;

int sum = 0;

while (i < 11)

{

sum = sum + i;

++i;

}

printf("%d\n",sum);

}

1. #include <stdio.h>

int main()

{

int i;

int sum = 0;

for (i = 0; i <= 10; ++i)

sum = sum + i;

printf("%d\n",sum);

}

1. #include <stdio.h>

int main()

{

int i = 0;

int sum = 0;

while (i < 11)

sum = sum + i++; printf("%d\n",sum);

}

1. #include <stdio.h>

int main()

{

int i = 0;

int sum = 0;

for(i = 0; i <= 10 ;)

sum = sum + i++; printf("%d\n",sum);

}

15.3

#include <stdio.h>

main()

{

int smallestNumber; int nextInput;

scanf("%d",&nextInput);

/\* We need to set the initial value of smallestNumber to something other than 0 as in the original code \*/

smallestNumber = nextInput;

while (nextInput != 1) {

if (nextInput < smallestNumber) smallestNumber = nextInput;

scanf("%d",&nextInput);

}

if (smallestNumber != 1)

printf("The smallest number is %d\n",smallestNumber); else

printf("No numbers entered.\n");

}

15.5

a. (2,3) (2,4) (3,4) (2,5) (3,5) (4,5) (2,6) (3,6) (4,6) (5,6)

1. 22
2. The code can be made more efficient by changing the inner loop to: for (j = 2; j < (i/2); j++)

Doing so reduces the number of calls made to IsDivisibleBy by half.

15.7

The program, as is, allows someone to purchase a ticket without first making a reservation.

The following program accepts only 32 reservations for the 10 available seats. A reservation is required before a ticket can be purchased.

#include <stdio.h> #define SEATS 10

#define MAX\_RESERVATIONS 32

int main()

{

int seatsAvailable = SEATS; char request;

int number;

int resStatus = 0; int resNumber = 0;

do { scanf("%c",&request);

if (request == 'R') {

if (seatsAvailable && resNumber < MAX\_RESERVATIONS) { printf("Reservation Approved\n");

printf("Your reservation number is %d\n", resNumber); resNumber++;

}

else

printf("Sorry, flight fully booked\n");

}

if (request == 'P') {

printf("Enter reservation number to confirm purchase : "); scanf("%d", &number);

if ((number >= resNumber) || /\* Invalid number \*/ (resStatus & (1 << number))) /\* Already purchased \*/

printf("Invalid reservation number. Purchase denied.\n"); else {

resStatus = resStatus | (1 << number); seatsAvailable--;

printf("Ticket Purchased!\n");

}

}

} while (request != 'X');

printf("Done! %d seats not sold\n", seatsAvailable);

}