**Practical 9 – refer to Topics 17 and 18**

**Part A (Understanding Concepts)**

1. Given the program below, what is the output if the input is the following?

(a) 5 (b) P (c) (Tab key) (d) g (e) ?

|  |
| --- |
| #include <iostream>  #include <cctype> // for character manipulation functions  using namespace std;  int main(void)  {  char ch;    cout << "Enter a character: ";  ch = getchar();  if (isalnum(ch))  {  cout << "Alphanumeric character\n";  if (isalpha(ch))  {  if (islower(ch))  cout << "Lowercase letter\n";  else if (isupper(ch))  cout << "Uppercase letter\n";  }  else if (isdigit(ch))  cout << "Digit\n";  }  else if (isspace(ch))  cout << "Whitespace\n";  else  cout << "Some other character\n";  return 0;  } |

|  |  |
| --- | --- |
| (a) What is the output if the input is the following? | |
| 1. 5 | Alphanumeric character  Digit |
| 1. P | Alphanumeric character  Uppercase letter |
| 1. (Tab key) | Whitespace |
| 1. g | Alphanumeric character  Lowercase letter |
| 1. ? | Some other character |
| (b) Why code ch = getchar();is usedinstead ofcin>>ch; ? | Because object cin cannot be used to read whitespace characters, for example space, tab, newline, etc. Whereas getchar() function can read all types of characters. |

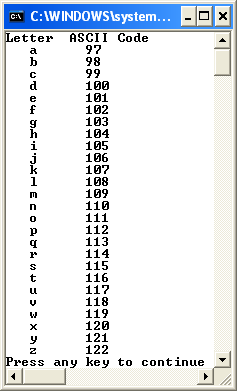
1. Given the program below, what is the output if the input is the following?

(a) h (b) Y (c) 3

|  |
| --- |
| #include <iostream>  #include <cctype> // for character manipulation functions  using namespace std;  int main(void)  {  char ch\_in, ch\_out;    cout << "Enter a character: ";  cin >> ch\_in;  ch\_out = tolower(ch\_in);  cout << "tolower of " << ch\_in << " returns " << ch\_out << endl;  ch\_out = toupper(ch\_in);  cout << "toupper of " << ch\_in << " returns " << ch\_out << endl;  return 0;  } |

|  |  |
| --- | --- |
| (a) What is the output if the input is the following? | |
| 1. h | tolower of h returns h  toupper of h returns H |
| 1. Y | tolower of Y returns y  toupper of Y returns Y |
| 1. 3 | tolower of 3 returns 3  toupper of 3 returns 3 |
| (b) What will be the output if a non-alphabet character is sent to function tolower() or toupper() ? | The tolower() or toupper() function will return back the unchanged non-alphabet as output. |

1. Write a program to display a table showing the characters from *a* to *z* together with their ASCII codes. A sample output is as follows:



#include <iostream>

using namespace std;

int main(void)

{

char ch;

cout << "Letter\tASCII Code\n";

for (ch = 'a'; ch <= 'z'; ch++)

cout << " " << ch << "\t " << (int)ch << endl;

return 0;

}

1. What is the output of the following program?

|  |
| --- |
| #include <iostream>  #include <cstring> // for string manipulation functions  using namespace std;  int main(void)  {  char s1[15], s2[5];  strcpy(s1, "good");  strcpy(s2, " job");    cout << "s1 length: " << strlen(s1) << endl;  cout << "s2 length: " << strlen(s2) << endl;  cout << "s1: " << s1 << endl;  cout << "s2 from 2nd element: " << &s2[1] << endl;  strcat(s1, s2);  cout << "s1: " << s1 << endl;  strncat(s1, s2, 3);  cout << "s1: " << s1 << endl;  return 0;  } |

s1 length: 4

s2 length: 4

s1: good

s2 from 2nd element: job

s1: good job

s1: good job jo

Press any key to continue . . .

1. Given the declarations below, what is the output of the following statements?

char s1[10] = "good";

char s2[10] = "goose";

char s3[10] = "goodies";

1. cout << strcmp(s1, s2) << endl;  
   -1
2. cout << strcmp(s2, s3) << endl;  
   1
3. cout << strncmp(s1, s3, 4) << endl;

0

1. Write statements for the following:
2. Declare an array named table with 5 rows and 3 columns to store 15 integers.
3. Assign all the elements to the value 0 using nested loops.
4. Assign the value 1 to all the elements in the first row using a loop.
5. Assign the value -1 to all the elements in the last column using a loop.
6. Compute and display the total of all the elements using nested loops.

|  |
| --- |
| int table[5][3]; |
| for (row = 0; row < 5; row++)  {  for (column = 0; column < 3; column++)  table[row][col] = 0; |
| for (col = 0; col < 3; col++)  table[0][col] = 1; |
| for (row = 0; row < 5; row++)  table[row][2] = -1; |
| total = 0;  for (row = 0; row < 5; row++)  for (col = 0; col < 3; col++)  total += table[row][col];  cout << “Total is “ << total << endl; |

1. What is the output of the following program fragment?

|  |
| --- |
| int table[6][6];  for (int row = 0; row < 6; row++)  for (int col = 0; col < 6; col++)  if (row == col)  table[row][col] = 0;  else if (row > col)  table[row][col] = -1;  else  table[row][col] = 1;  for (int row = 0; row < 6; row++)  {  for (int col = 0; col < 6; col++)  cout << setw(3) << table[row][col];  cout << endl;  } |

0 1 1 1 1 1

-1 0 1 1 1 1

-1 -1 0 1 1 1

-1 -1 -1 0 1 1

-1 -1 -1 -1 0 1

-1 -1 -1 -1 -1 0

**Part B (Programming Exercises)**

1. Write a program that accepts a character from the user and displays the next character in the alphabet. Your program should only do this if the input is an alphabetic character. If the character entered is *z* (or *Z*), display *a* (or *A*).

#include <iostream>

using namespace std;

int main(void)

{

char in, out;

cout << "enter a character: ";

cin >> in;

if (isalpha(in))

{

if (in != 'z' && in != 'Z')

out = in + 1;

else

if (in == 'z')

out == 'a';

else

out = 'A';

cout << "Next chac is: " << out << endl;

}

else

cout << "This is not a letter" << endl;

return 0;

}

1. Write a program that asks the user to enter a word. The program will display the word backward. Assume the word will not exceed 20 characters. Note: The program should display the characters one by one starting from the last character.

Sample run:

Enter a word: program

The word printed backwards: margorp  
  
#include <iostream>

using namespace std;

int main(void)

{

char word[20];

cout << "enter a word: ";

cin >> word;

cout << "The word printed backwards is: " << endl;

int len = strlen(word);

for (int i = len - 1; i >= 0; i--)

cout << word[i];

cout << endl;

return 0;

}

1. Modify the program in question 3 to ask the user to enter a *sentence*. The program will display the sentence backward. Assume the sentence will not exceed 50 characters. Note: Use function *gets* to input the sentence and store in an array.

Sample run:

Enter a sentence: This is the sentence.

The sentence printed backwards: .ecnetnes eht si sihT  
  
#include <iostream>

using namespace std;

int main(void)

{

char word[20];

cout << "enter a word: ";

cin >> word;

cout << "The word printed backwards is: " << endl;

int len = strlen(word);

for (int i = len - 1; i >= 0; i--)

cout << word[i];

cout << endl;

return 0;

}

1. Write and test a function named *count\_upper* that has a string parameter. The function counts and returns the number of uppercase letters in the string.
2. Write and test a function named *search\_char* that has 2 parameters: a string (i.e. an array of characters) and a single character. The function counts and returns the number of times the character appears in the string.
3. Write and test a function that has a two dimensional array and its number of rows as parameters. The function computes and returns the *largest* of all the elements of the array.

**Part C (Self-Review / Revision)**

1. What is the purpose of each of the following character manipulation functions?

isalpha, isdigit, isalnum, isupper, islower, isspace, toupper, tolower.

1. How is a character stored in the computer?   
   stored in a pc using its ASCII code
2. How is a string stored in the computer?  
   stored in pc using an array of characters and delimited by a null character
3. What is the difference between reading a string using *cin* and *cin.get()*?  
   when reading a string using cin, a whitespace indicates the end of the string, so cin cannot be used to read a string containing spaces
4. What is the purpose of each of the following string manipulation functions?

strlen, returns the length of the string

strcpy, copies content of one string to another

strncpy, copies content of one string to another for a specified no of characters

strcmp, compare 2 strings

strcat, appends or concatenates one string at the end of another string

strncat. Appends the specified no of characters of one string at the end of another string

1. What is the general format to declare a two-dimensional array?   
   element-type array-name[noofrows][noofcolumns]
2. How do you access an element of a two-dimensional array?  
   array-name[row-index][column-index]

**Part D (Practice Exercises)**

1. Write a program that accepts a letter and displays the position of the letter in the alphabet. For example, letter *a* has position 1, *b* position 2, and so on. The program should be able to handle both upper- and lowercase letters. If the character entered is not alphabetic, the program should display an error message and stop.

#include <iostream>

#include <cctype>

#define A\_ASCII\_CODE 97

using namespace std;

int main(void)

{

char ch;

int position;

cout << "Enter a letter: ";

cin >> ch;

if (isalpha(ch))

{

ch = tolower(ch);

position = ch - A\_ASCII\_CODE + 1;

cout << "The position in the alphabet is : " << position << endl;

}

else

cout << "This is not a letter\n";

return 0;

}

1. Write and test a function named *convert\_case* that has an array of characters as a parameter. The function converts all the letters in the word to uppercase letters.

#include <iostream>

#include <cctype>

#define A\_ASCII\_CODE 97

using namespace std;

void convert\_case(char word[]);

int main(void)

{

char word[20];

cout << "Enter a word: ";

cin >> word;

convert\_case(word);

cout << "The word is now : " << word << endl;

return 0;

}

void convert\_case(char word[])

{

int len = strlen(word);

for (int i = 0; i < len; i++)

word[i] = toupper(word[i]);

return;

}

1. Write and test a function *reverse\_word* that has a string parameter. The function reverses the letters in the string. The program should perform the following to exchange the first and last characters in the string:
2. Copy the first character of the string to a temporary variable.

g

o

o

g

l

e

\0

?

?

g

0 1 2 3 4 5 6 7 8

temp

1. Copy the last character to the first position in the string.

e

o

o

g

l

e

\0

?

?

g

0 1 2 3 4 5 6 7 8

temp

1. Copy the character in the temporary variable to the last character position in the string.

e

o

o

g

l

g

\0

?

?

g

0 1 2 3 4 5 6 7 8

temp

Repeat the steps to exchange the second and second-last characters, third and third-last characters, and so on.  
  
#include <iostream>

#include <cstring>

using namespace std;

void reverse\_word(char word[]);

int main(void)

{

char word[20];

cout << "Enter a word: ";

cin >> word;

reverse\_word(word);

cout << "The word reversed is now : " << word << endl;

return 0;

}

void reverse\_word(char word[])

{

int half\_len, front, back;

char temp;

back = strlen(word) - 1;

half\_len = strlen(word) / 2;

for (front = 0; front < half\_len; front++, back--)

{

temp = word[front];

word[front] = word[back];

word[back] = temp;

}

}

1. Write and test a function that has a two dimensional array and its number of rows as parameters. The function computes and returns the *total* of all the elements of the array.

#include <iostream>

using namespace std;

int find\_total(int table[][4], int rows);

int main(void)

{

int table[3][4] =

{

{ 1,1,1,1 },

{ 2,2,2,2 },

{ 3,3,3,3 }

};

int total = find\_total(table, 3);

cout << "Total is " << total << endl;

return 0;

}

int find\_total(int table[][4], int rows)

{

int total = 0;

for (int i = 0; i < rows; i++)

for (int j = 0; j < 4; j++)

total += table[i][j];

return total;

}

1. Write and test a function that has a two dimensional array and its number of rows as parameters. The function computes and returns the *smallest* of all the elements of the array.

#include <iostream>

using namespace std;

int find\_smallest(int table[][4], int rows);

int main(void)

{

int table[3][4] =

{

{ 1,1,1,1 },

{ 2,2,2,2 },

{ 3,3,3,3 }

};

int smallest = find\_smallest(table, 3);

cout << "Smallest is " << smallest << endl;

return 0;

}

int find\_smallest(int table[][4], int rows)

{

int smallest = table[0][0];

for (int i = 0; i < rows; i++)

for (int j = 0; j < 4; j++)

if (smallest > table[i][j])

smallest = table[i][j];

return smallest;

}