| UNIT-OL | The Lift I was a second of the comment of the comme |
|---------|--|
| COU | FUNCTION (A) PAGE: |
| | Function: |
| | of 1. ston a organim. It is also a |
| | ganup of statement that assuper |
| | to perform a task. |
| | |
| رب | A function always return value except |
| | void data type. |
| | of material ! |
| 0 | refurn-type function-name (parameter list) |
| ,03 | { |
|) | statement? |
| | } |
| | |
| * | Type of function: |
| | Par delined by ablan |
| 0 | Pre-defined function User-defined function |
| | |
| SO | Pre-defined function/Library function/ Inbuilt function |
| (0) | Inhuilt tunction |
| <u></u> | The tunction which meaning is already - |
| | The function which meaning is already—defined in Clanguage library is known—as pre-defined function or library function.— |
| | as now defined tunction ox library function. |
| | Exi |
| | getch (), claser (), streen (), streat (), |
| | getch(), clrscr(), strlen(), strcat(), - strcmp(), strcpy(), strupr(), gets(), - |
| | outs () etc. |
| 250 | 250 |
| | |

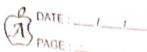
| | PAGE: |
|--|---|
| ai | User-defined function: User-defined function is that function which is defined by the user. |
| ب | User-defined function is that function which is |
| STATE OF THE PARTY | defined by the user. |
| | add (), sub (), mul (), div(), etc. |
| | , mu (), and otc. |
| (i) | There are four types of user defined function! |
| رنن | No return with argument passing |
| _aii | Resurn with no arnument no seine |
| CÁD | Return with argument passing |
| | Assument |
| | No orgument |
| | (Calling) |
| | function (called function) |
| | |
| | Return |
| | No return |
| 31 | 3 |
| | Note: - Main function is always calling |
| | Note: - Main function is always calling function & sub function imay be calling function or may be called function. |
| | function or may be called function. |
| | |
| | |
| | |
| | 25° |
| | 31,928 |
| | |

| (i) No return with no argument couling The arguments to the called function described also called function described return the any values to the calling function. Is known as no return with an argument passing. Ex:- Living a program to find the addition of the number with function. ## include statich? ## include stat | | CATE:/ |
|--|--|--|
| Inherever colling function doesn't rave the arguments to the collect function be also called function doesn't return the any values to the colling function is known as no return with no argument passing. Ex:- Livite a program to find the addition of two number using function: ## include statich > ## include statich > whid addit int a, b, c; printf ("In enter the value of a & h"); scant (" '' I d' d' d' a & h); (= a + h; printf ("In sum = 1.d' c); void main () closer(); add (); | en | A) PAGE: |
| Inherever colling function doesn't gave the arguments to the colled function be also called function doesn't return the any values to the colling function is known as no return with no argument passing. Ex: bluite a program to find the addition of two number using function: ## include statich > ## include scanich > whid addition of the point ("In enter the value of a & h"); scant ("In enter the value of a & h"); scant ("In sum = I.d" c); vaid main () creer(); add(); | (6) | No return with the argument possing: |
| the arguments to the called function a also called tenction deeper action the any values to the colling function, is known as no return with no argument passing. Ex:- Limite a program to find the addition of two number using tunction: ## include (conin.h) while abid! absorbed! int a, b, c! printf ("In enter the value of a & h"); scart (" Y. J. Y. do & a, k, h); (= a + h; printf ("In sum = Y. d', c); Void main!) (drscr(); add(); | | |
| also Falled function doesn't return the any values to the colling function is known as no return with no argument passing. Ex:- Living a program to find the addition of two number wing function: # include (someth) which addit int a, b, c; printf ("In enter the value of a le h"); scort ("Y.d Y.d of le n, le h); c= a + h; printf ("In sum = Y.d", c); Void main () clyser(); add (); | | Whenever colling function doesn't gave |
| any values to the colling function, is known as no return with no argument passing. Ex:- Write a precupit to find the addition of two number wing function: ## include statich ## include sconin-t wid addition ## include sconin-t wide sconin-t ## include sconin-t wide sconin-t ## include sconi | | |
| as "no return with no argument passing. Ex:- brite a program to find the addition of two number wing function. ## include settlich > ## include second. > void addi) about to print (" In enter the value of a & h"); seant (" Y. d Y. d & ka & h); c a + h; void main () Void main () Urser(); add (); | The second secon | |
| Lexi- White a program to find the addition of two number using function: ## include Station > ## include Sta | | |
| # include Soldich > # include Soldich > # include Sconich > void Goldi) int a, b, c : printf (" In enter the value of a & h"); sconf (" Y. J Y. J. S. & a, & h); (= a+h; printf (" Y. Sum = Y. d", c); void main () clrscr(); add (); | | Exe- |
| # include Soldich > # include Soldich > # include Sconich > void Goldi) int a, b, c : printf (" In enter the value of a & h"); sconf (" Y. J Y. J. S. & a, & h); (= a+h; printf (" Y. Sum = Y. d", c); void main () clrscr(); add (); | | Write a program to find the addition of |
| # include (statich) # include (conin.h) void addi) int a,b,c: printf ("In enter the value of a & h"); seart (" Y.d Y.de & ka. & h); c= a+h; printf ("Yh sum = Y.d' c); void main () closer(); add (); | 250. | two number wine function. |
| # include (conjo.h) void addil) checiptes int a, b, c; printf ("In enter the value of a le h"); seant (" Y. d Y. de le la | 3,0 | |
| int a, b, c; printf (" n enter flow value of a & h"); scant (" Y. d Y. do & ka. kh); c= a+h; printf (" n sum = Y. d" c); void main () clyser(); add (); | | # include Xotdio.h) |
| int a, b, c; printf ("In enter the value of a & b"); scant (" Y. J Y. de & & a & b); c= a+b; printf ("In sum = Y. d', c); Void (main 1) clrscr(); add (); | State of the state | # include (conin.h) |
| printf (" n enter the value of a & h"); scant (" Y. d Y. do' & ka. & h); c= a+h; printf (" n sum = "I. d" c); void main () drscr(); add (); | State of the state | upid coddi) |
| printf (" n enter the value of a & h"); scant (" Y. d Y. do' & ka. & h); c= a+h; printf (" n sum = "I. d" c); void main () drscr(); add (); | and the second | 7.0 |
| printf (" n enter the value of a le h"); scant (" Y. d Y. de le la | | ebecineta de la companya della compa |
| C= a+h; printf ("In sum = 1.d", c); | | $\frac{int a, b, c}{n}$ |
| C= a+h; printf ("In sum = "I.d", c); | | printf (In enter flor value it a & b) |
| void main U closer(): add (); | ,60 | seart (1.21.25 ka. kb); |
| void main U closer(): add (); | 3 | c = a + b; |
| drscr(); | | Print (m sum = 1.d c); |
| drscr(); | List of the second | |
| add (); | marida A representativo | Void main () |
| add (); | the same of the sa | 10000 |
| | To the second | |
| getten U | | |
| | -07 | getten U; |
| 1,65 | 492 11 | 1 26,00 |
| 2, 12, | | 1,2, |

No return with aroument passing. In this type of function, calling function will passes the arguments to the called function but called function will not neturn any value to the WAP to find the addition of two number using function. # include (stdio.h) formal argument # include (conio.h) void add (int x, int y) 7= x + y; orinf ("In sum = 7.d", 7); void main () Claser (); printf (" In enter the two values"); scant ("Y.d 7.d", ka, lb); add (a.B); - Actual argument getch ();

| / 1 | DATE | |
|-----|------|------|
| AS | PAGE | |

| | A DATE:/_/_ PAGE:/_/ |
|-------|--|
| 0 | |
| Citio | Return with no argument passing: |
| 3 | m 11 a 11 |
| | 11 11 11 11 11 11 11 11 11 11 11 11 11 |
| | Lalle View high College |
| | noturn a value to the colling function. |
| | Ev = |
| | WAP to find the addition of two no. using |
| - | function. |
| | #include < stdio.h> |
| 20. | #include (conso.h) |
| 310 | voidemainted int add () |
| 3 | |
| | intab, c; |
| | printf ("In enter the value of a & b"); |
| | 26074 ("Y 1 7.1" & 25 Pi |
| | C = a + b; |
| | return C; |
| | } |
| | unid main () |
| .65 | { |
| 3 | ints; |
| | drscr(); |
| | s=addU; |
| | printf (" n sum= 1.d", s) |
| | punge in sum- 1.0, sp |
| | getch (); |
| | |
| | |
| 250 | 250 |
| 240 | Chicago Chicag |
| | |



Priototype function: It is a dype of function, in which the name of substanction is define below the header file and above the main function and the wholes sub program is defined | written below the main function is known as prototype function. WAP to find the addition of two no's # include (stdip. R) It include (conio. h) void add () /xprotodype function x/ void main () drscru; add U; getch (); void add U pointf ("1.d", lea, & b); printf ("In sum= 1.d", c);

| ng. | A DATE:/_/_ PAGE: |
|--|--|
| 8. | WAP to colculate the factorial of any |
| 2 | given no. & also check the even or odd |
| | WAP to calculate the factorial of any given no. & also check the even or odd no. using prototype function. |
| ted to the management of distribution in an interpretation | |
| 1996/fillionische Schroe Mannie des plagenomes names | # 63 |
| result-six-vite-finate (MC), the core convolute constitution and | # |
| | |
| | |
| 25 | 75 |
| 3 | 3 |
| | 9/4 |
| | |
| | <u>(6)</u> |
| | |
| | |
| | |
| 200 | |
| 2(0) | 269 |
| 2 | |
| | 0 |
| | 60/ |
| | . 6 |
| | |
| | |
| | |
| 250 | 250 250 |
| 310 | City. |

| | DATE: |
|---|--|
| | PAGE: |
| 30 | |
| | |
| | |
| AND DESCRIPTION OF THE PERSON NAMED IN COLUMN | - CO/ |
| - | C ₃ |
| | |
| | |
| | |
| | |
| 193 | |
| 5 | 3 |
| | Recursion: |
| | Recursion is a type of function whenever a function call itself is known as recursion. |
| , | a function call itself is known as many line |
| | Or, |
| | we know that a function calls another |
| | function or sub function whenever a function |
| | called if itself its known as recursion. |
| S | Fx: |
| 10 | WAP to find a factorial of any given no. |
|) | using recursion. |
| | using recursion. |
| | # include (stdio.h) |
| | # include (conjo.h) |
| | int fact (int n); |
| | void main () |
| | (1010 11 (211) |
| | \(\(\text{U18} \) \(\text{17} \) \(\text{17} \) |
| | $\begin{cases} \\ \\ \\ \\ \end{cases}$ int n, t ; |

| ell' | PAGE: |
|--|---|
| 93 | clyscy(); |
| | claser(); printf-("In enter the value of n"); scanf ("I.d", & n); f = fact (n); printf-("In factorial = 7.d", f); |
| | scant ().d", & n); |
| | 1 = gact (n); |
| electronic page 1 and 1 de l'establishe de l'e | printy ("In factorial = 7.0, f); |
| | getch(); |
| Control of the second of the s | 50 |
| | int fact (Int n) |
| | |
| 92 | int v=1; 55 |
| | y(n=0) |
| The second secon | return V; |
| | else. |
| | van x fact (n-1); |
| | return V; |
| | |
| ~ | |
| (0) | Enter the value of n. |
| 92 | Enter the value of 1) |
| | 50 |
| | $5 = 0 \times 1$ $V = 5 \times fact(4)$ |
| | 4 = c = 0 × |
| | |
| | $V = 5 \times 4 \times fact (3)$ |
| | 3==0 X |
| | V= 5x4x3x fact (2) |
| - 6 | 2 = = 0 X |
| 921 | .63 |

| | | - ASS |
|-------|---|-------|
| en | PAGE:/ | 1 1 |
| ii) | Repister Storage Class: | |
| 5 | The register storage class is used | to |
| , | define local variables that should be | |
| | stored in a register instead of RAM. | |
| | i.e. means the variable has a maximum | |
| | size equal to the register size & can't | _ |
| | have the uniary AND operator applied | |
| | to it. | |
| | Ex! | |
| 100 | 1 1010 | _ |
| 5 | rugister int x; | _ |
| | | * |
| | St 118101 00 | |
| - 11) | Static Storage Class: | _ |
| - | The static storage class instructs the | _ |
| | compiler to keep a local variable in | _ |
| | existance during the life time of the | _ |
| 6 | program instead of creating and destroy | ng |
| 23 | ack time is comes into and ones | U |
| 5 | out of scope, therefore making local | - |
| | priable Static allows them to maintain | |
| 7 | heir values between function calls. | , |
| th | The static modify may also applied to | |
| | In Clarente. | |
| | In C language, when static is used | |
| 0 | Towney variable, il cause only on a consu | |
| | that member to share by all the | , |
| 1,05 | .,05 | |
| 3) | | |
| T. | | |

| | PAGE: |
|-------|--|
| 6, | object of its class. |
| 9 | Ex:- |
| | #- 0000 |
| | void abel); |
| i i | static int count: 5; /x bibbal variable x/ |
| | Void main () |
| | } |
| | while (count) |
| | |
| 300 | abc (); |
| - | |
| | neturn O; |
| | yoid abc () |
| | \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\ |
| | static int i=5; /* local variable.*/ |
| | l ++ |
| | printf ("In i is Yed and count is Yed", i, count |
| 40° | |
| 3 | 3 |
| _iv | Extern Storage Class! |
| | The storage class is used to give a reference of a plobal variable i.e. visible to all the propran |
| 110 | of a global variable i.e. visible to all the propran |
| | files. |
| 150 : | When we use 'extern', the variable can't be |
| | storage location that has been previously defined. |
| 1120 | The rest of the second providing defined |
| NO. | |
| | |