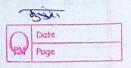
Exception Handling



77

Exceptions are the conditions that arises infrequently and unempercedly and which are generally different from Common perogramming errors such as syntam errors.

There exceptions may lead to crash of the program and may cause abnormal exit from the program.

In a program, exception may be Fraised due to several reasons such as low disk space, low free main memosz, divide by zero condition, pop form an empty stack, put to a full stack, memoszy read-write enought.

Exception handling: - is a mechanism that separates the detection and handling of circumstantial exceptional flow.

CHT provides a mechanism to handle such enceptions.

Which are generated due to some discrepencies in the program. This mechanism is called as try-throw—

Catch mechanism. In this mechanism, the section of the code which may generate an enception, is kept inside—

the try' section. If an enception is caught, then using the keyword throw, the Control is pushed to a section called as 'Catch' where appropriate action is acte taken to handle the enception. For enception, and the situation of 'divide by tero' enception, then program may crash.

int main () 2 int a= 100 int b= 10; int c= a-b' but q. cout [(d', 17 hore program will create an exception, divide by their Hence, the program will enough and will be abnormally arented. So, Let we apply here the try-throw- (atch mechanism. So, now it will look as follows int maine) d int a= 10; mt c = a-b', if (c1=0) Carch (int m) contic enception is handled; now, the program will not undergo the abnormal aboution. Hore, in the 'try' section, the exception is caught and wing the statement 'throw c; the a control in being switched to the Catch section & exception is handled.

Here inthe state ment throw (;), (is an integer. Here, in the Catch' section, this c is being received by by which is also an integr. Durie of poplace of m' is of flood-type then due to mismatch The for control will not be able to come under the catch section sand program will crash. So, we should roop in he careful while throwing an enception that variable data-type must be marched marched. multiple catch Statement within a particular try section, exceptions may be raised to due to several reasons and hence, for each reason, we may need a separate Catch section. This situation leads to the need of multiple catch Fun enample: int main () int 0= 10; float b = 20.5 chan (= 'D') if (condition 1) throw a; else if (Condition 2) Amow b. else throw C: Catch (int m) ¿ coud cc" Int enception handled; } Catch (float n) of Aprova (c. floor Enception pandled.)

(arch (chas P) { coulce "chan enception handled";) Here, depending upon the throw' statement, a relevant 'Carch' Action will be invoked. Here, implicit type Conversion is not moidone. Even after loss taking partiple farch statement, it is possible that for Sometime, it is also possible that exall type of throw' statements can be handled by a single Catch xction. So, in this can, multiple catch statement is not orequired. For such care, (atch (...) is und. int main () 1 int a = 10; float 5=205; chan c = 'D'; if (Condition) throw a; ebuil (conditions) of throw b; elk & throw C; (cow LC all enceptions are handled here")

Rethrow of enceptions when a catch statement not only handles a particula enception but also throws the same enception to another Catch Statement which is part of next enclosing +27- Catch sequence. must of the firmer, enceptions are conglet not caught in main function but in deep nexted function calls. e.g. majne); func); d'mune); Sur it is pussible that enception is caught within the munc) function. So, in order to propogate the enception to the main function we need to stethnow the enception. main!) try of function and hard to the same trans crown sicher translable dotos 2 (atch (int x) d coul CC" enception congret in main"; & House void Catch (int y)

{ cout (enception (aught in fun ;

throw;

void gunc) munc); catch (int 2) Cout CC" exception cought ingun"; thouse men; In the above program, 1st fun() will be called, then gune and and Inside the funct, gune will be called . Inside the gun(), munc) will be called. Let, within the omunc), exception is caught and being thrown by the statement throw m' This throw' will be received by the catch statement within gune function. This 'Carch' statement lot prints & "exception caught in gun' and hen rethrows the exception which will be received by catch' Statement inside the funcl. Further, this catch statement prints " enception caught in fun and then, redhows the enception to the catch statement within the main function. Finally, this catch statement prints exception (aught "in main" and not nethows further. In this way, an enception naised in the mun() function propagates to the main function.