BITS Pilani
Pilani Campus

Nalanda
The On-Campus Learning Management System

Navigation
Administration
Important Links
Contact Us
Events
Calendar
Upcoming events
MOBILE APP LOGIN

Nalanda

H

Dashboard ► My courses ► 11461001318_CS F363 ► General ► Announcements ►
Stage 2: Marks distribution, test cases, evaluatio...

Search forums

Announcements

Stage 2: Marks distribution, test cases, evaluation criteria and marks list

◄ Mid semester test: Marks list, solution and evaluation criteria

Display replies in nested form

Stage 2: Marks distribution, test cases, evaluation criteria and marks list
by Vandana Agarwal . . - Tuesday, 14 July 2020, 8:30 AM

Code evaluation:

The credit has been given to the tested code as per the following criteria. Marks are awarded on the correctness of the results.

The test cases used in evaluation are attached herewith for your reference and self-assessment of your code. The expected errors of the semantic analysis testcases u1-u10 are listed alongside the code. The expected output of the code generation test cases v1-v11 are mentioned below the code in commented lines. The teams are advised to test the working of their code. Inform me immediately, if you observe any discrepancy in the test cases.

I have observed that a few codes have compatibility issues and many modules are suffering from segmentation faults. If you have obtained marks less than your expectations, please do not panic. Most zeros are due to the segmentation faults your code displayed on my system in that module. The recheck requests should be uploaded through Nalanda at the new link open from now till tomorrow evening 6:00 p.m.

A sum total of 25% marks has been deducted from the total marks obtained by the teams, if they availed more than one lifeline.

Navigation
Administration
nportant Links
Contact Us
Events
Calendar
Upcoming events
MOBILE APP LOGIN

Please feel free to write to me any issue you face regarding your marks.

Test cases used for evaluation: The test cases used for evaluation are the minor variations of the test cases provided earlier. The purpose of each test case is mentioned at the top of its code. The marks are awarded only for the tested code. A total of 22 test cases are used for evaluation. The implementation of Abstract Syntax Tree, symbol table, type checking and semantic analysis is evaluated using the test cases u1 to u10 using the line number wise reporting of semantic errors. Also, the credit for printing of AST, memory compression computation, symbol table (in the prescribed format), activation record sizes and printing of information about the static and dynamic arrays is given separately using a separate test case s1.txt for evaluation. The code generation test cases v1 to v11 contain semantic error free code and are used for generating code.asm file using your compiler, which is executed using NASM as per the suggested instructions by each team.

Marks Distribution

The marks distribution for all is as follows

[A] Printing of details (as asked through driver options 3-7): 8 Marks

[A1-A6 use test case s1.txt]

A1. Paper work on AST rules : 1M

A2. AST printing : 2M

A3. Computation of compression ratio for used memory: 1M

A4. Symbol table printing: 2M

A5. Activation record size printing:1 M

A6. Printing of information about static and dynamic arrays: 1M

[B] Semantic analysis : 24 Marks

B1. Variable is declared or not/ is re-declared (nesting level:1, only primitive data types, test case u1.txt): 1M

B2. Variable is declared or not/ is re-declared (nesting level:1, primitive data types and static arrays, test case u2.txt): 1M

B3. Variable is declared or not/ is re-declared (nesting level:2,3, only primitive data types, test case u3.txt): 1M

B4. Variable is declared or not/ is re-declared (nesting level:2,3, primitive data types and static arrays, test case u4.txt): 1M

B5. Static bound checking and type checking (nesting level:2,3, primitive data types and static arrays, test case u5.txt): 4M

B6. Number and types of input and output parameters (nesting level:2,3, primitive data types and static arrays, test case u6.txt): 4M

B7. Function precedence, declaration and redundancy, assignment to output parameters (nesting level:2,3, primitive data types and static arrays, test case u7.txt): 4M

B8. Function overloading and recursion (nesting level:2,3, primitive data types and static arrays, test case u8.txt): 2M

B9. Execution flow statements- switch, while and for (nesting level:2,3, primitive data types and static arrays, with all errors as above, test case u9.txt): 4M

Navigation
Administration
Important Links
Contact Us
Events
Calendar
Upcoming events
MOBILE APP LOGIN

B10. Execution flow statements- switch, while and for (nesting level:2,3, primitive data types and static arrays, independent test case, test case u10.txt): 2M

[C] Code Generation: 13 Marks

C1. Simple expression computation involving integer and real numbers (nesting level:1, single driver module, test case v1.txt): 1M

C2. Simple expression computation involving integer numbers and Boolean values (nesting level:1, single driver module, test case v2.txt): 1M

C3. Simple expression computation involving integer numbers, Boolean values and static arrays (nesting level:1, single driver module, test case v3.txt): 1M

C4. Simple expression computation involving integer numbers, Boolean values and static arrays (nesting level:2, single driver module, test case v4.txt): 1M

C5. Expression computation involving static arrays in a loop (nesting level:2, single driver module, test case v5.txt): 1M

C6. Dynamic bound checking and Run time error reporting using static arrays (nesting level:2, single driver module, test case v6.txt): 1M

C7. Dynamic type checking with static and dynamic arrays (nesting level:1, single driver module, test case v7.txt): 1M

C8. Expression computation using switch statement (nesting level:2, single driver module, test case v8.txt): 0.5 M

C9. Expression computation using switch statement under a for loop (nesting level:3, single driver module, test case v9.txt): 0.5 M

C10. Function call implementation using parameters as primitive data type (nesting level:1, driver and one function module, test case v10.txt): 1 M

C11. Function call implementation using array parameters – usage of both static and dynamic arrays (nesting level:2, driver and one function module, test case v11.txt): 4 M

Evaluation criteria:

Paper work [A1]: Any team who submitted AST rules has been given 1 mark.

Printing of details [A2-A6]: Partial marks are given for partially correct output (granular scale of 0.5)

Semantic analysis test cases [B1-B10]: The reported line numbers and associated error messages of all detected errors in the test case code were verified. The errors specific to the purpose of the test case, if reported correctly, are credited, provided other existing errors are also detected appropriately qualitatively and quantitatively. If the lines where it is mentioned specifically that there is no error in that line, are also detected by your compiler and reported as semantic errors, then marks were proportionately deducted as well. Marks are given in the granular sizes of 0.5 in u1-u10 evaluation.

Code generation test cases [C1-C11]: The expected output for the instances of the input given at the end of test cases v1-v11 are mentioned. Test cases v1-v7 were awarded 1 if the output is fully correct, and were awarded 0 if there is any wrong value in the output. Test cases v6 and v7 are for testing dynamic bound and type errors respectively and are not given any credit if the code execution does not report the error message appropriately after the necessary output prior to the error. Test cases v8 and v9 are evaluated for 0.5 marks each and there is no partial marking in these. Only the

Navigation
Administration
nportant Links
Contact Us
Events
Calendar
Upcoming events
MOBILE APP LOGIN

function implementation test cases v10 and v11 were credited for partial correctness. **For v11**, **1 mark** is given for printing the contents of the formal parameters after they receive values from actual array parameters and **1.5 marks** are for computing values of sum1, sum2 and sum3 (0.5 each) over all iterations. This tested the static and dynamic array parameters combination with local array parameters, both static and dynamic. The value of k computed at line 46 is credited for **0.5 mark**. The return of output parameters sum1, sum2 and sum3 into the actual parameters s1, s2, s3 correctly and printing of values in s1, s2 and s3 has been credited for **1 mark**.

Recheck request: Genuine recheck request for stage 2 code will be accepted till tomorrow evening 6:00 p.m. (Separate link provided on the main course page).

Stage 1 recheck status: stage 1 code of the teams who had given requests for recheck earlier, will be re-evaluated today and tomorrow. The updates will be communicated to the students by tomorrow evening.

Pre-Compre total: This will be uploaded by July 19, 2020, Sunday, 5:00 p.m.

Marks List

Compiler Construction (CS F363): Stage 2 marks (July 14, 2020)

		Printing of details as asked through options 3-7 (8 M)								Semar	
Group number	AST paper work (1)	AST printing (2)	Mem compn. (1)	Sym table printing (2)	Activation record details printing (1)	Static/ dynamic array details printing (1)	u1 (1)	u2 (1)	u3 (1)	u4 (1)	u
1	1	2	1	2	1	1	1	1	1	1	4
2	1	2	1	2	1	1	1	1	1	1	4
3	0	0	0	0	0	0	0	0	0	0	0
4	1	2	1	2	1	1	1	1	1	1	4
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0

Navigation
Administration
nportant Links
Contact Us
Events
Calendar
Upcoming events
MOBILE APP LOGIN

7	1	2	1	1	1	1	1	1	1	1	4
8	1	2	1	1	0.5	0.5	1	1	1	1	4
9	1	2	1	2	1	1	1	1	1	1	4
10	1	2	1	2	1	1	1	1	1	1	4
11	1	2	1	2	1	1	1	1	1	1	4
12	1	2	0	0	0	0	0	0	0	0	0
13	1	2	1	2	1	1	1	1	1	1	4
14	1	2	1	2	1	1	1	1	1	1	3
15	1	2	1	2	1	1	1	1	1	1	4
16	1	2	1	2	1	1	1	0	1	1	4
17	1	2	1	2	1	1	1	1	1	1	4
18	1	2	1	2	0	0.5	0	0	0	0	0
19	1	2	1	2	0	1	1	1	1	1	4
20	1	2	1	2	1	1	1	1	1	1	4
21	0	2	1	0	0	0	0.5	0	0.5	1	3
22	1	2	1	0	0	0	1	1	1	1	4

Navigation
Administration
nportant Links
Contact Us
Events
Calendar
Upcoming events
MOBILE APP LOGIN

23	1	2	1	1	1	0.5	1	1	1	1	4
24	0	2	1	2	1	1	1	1	1	1	4
25	1	2	1	2	1	1	1	1	1	1	4
26	1	2	1	0.5	0.5	0.5	1	0	1	1	4
27	0	0	0	0	0	0	0	0	0	0	0
28	1	2	1	2	1	1	1	1	1	1	4
29	1	2	1	1	0.5	0.5	1	1	1	1	4
30	1	2	1	2	1	1	1	1	1	1	4
31	1	2	1	2	1	1	1	1	1	1	4
32	1	2	1	2	1	1	1	1	1	1	4
33	1	2	1	1	1	1	1	1	1	1	4
34	1	2	1	2	1	1	0	0	0	0	0
35	1	2	1	2	1	1	1	1	1	1	4
36	1	2	1	1	1	1	1	1	1	1	0
37	1	2	1	2	1	1	1	1	1	1	4
38	0	0	0	0	0	0	0	0	0	0	0

Navigation
Administration
nportant Links
Contact Us
Events
Calendar
Upcoming events
MOBILE APP LOGIN

39	1	2	1	2	1	1	1	1	1	1	4
40	1	1	1	1	1	1	1	1	1	1	4
41	0	0	0	0	0	0	0	0	0	0	0
42	1	2	1	1	0	1	1	1	1	1	4
43	1	2	0	0	0	0	0	0	0	0	0
44	1	2	1	2	1	1	0	0.5	0	0.5	0
45	0	2	1	0	0	0	0	0	0	0	0
46	1	2	1	2	1	1	1	1	1	1	4
47	1	2	1	2	1	1	1	1	1	1	4
48	1	2	1	2	1	1	1	1	1	1	4
49	1	1	0.5	1	0.5	0.5	1	0	1	0	3
50	1	2	1	2	1	1	1	1	1	1	4
51	1	2	0	0	0	0	0	0	0	0	0
52	1	2	1	2	0.5	1	1	1	1	1	4
53	0	2	1	0	0	0	1	1	1	1	4
54	1	2	1	1	0.5	1	1	1	1	1	4

```
**************************************************************************************************
```

testcases.zip

Permalink

◄ Mid semester test: Marks list, solution and evaluation criteria

Contact us : nalanda@pilani.bits-pilani.ac.in, Location : Media Lab, Room Number : 3231, SDET, FD-III, Pilani Campus