

Vistula, IT Faculty, 2014

Algorithms and Complexity

Dmitry A. Zaitsev

<http://daze.ho.ua>

Tasks for practical work and laboratory training

I. TM

Construct a TM (a state diagram & a sequence of instructions) and debug it, for instance with <http://morphett.info/turing/turing.html>. Provide a series of test examples.

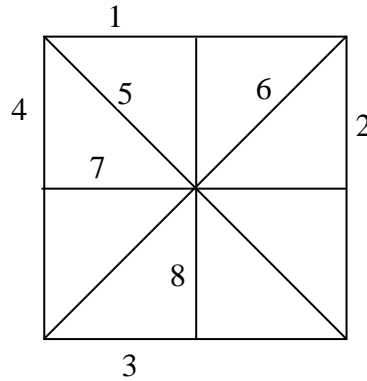
1. Subtraction in unary system
2. Division by 2 in unary system
3. Multiplication by 2 in unary system
4. Division by 3 in unary system
5. Multiplication by 3 in unary system
6. Erase symbol "a" within a word in alphabet {a,b,c}
7. Replace left and right parts of a word in alphabet {a,b}
8. Reverse a word in alphabet {a,b}
9. Addition in binary system
10. Subtraction in binary system
11. Interleave two words in alphabet {a,b}
12. Division by 4 in binary system
13. Remainder of division by 5 in unary system
14. Erase repeated "a" within a word in alphabet {a,b,c}
15. Replace odd and even letters of a word in alphabet {a,b}
16. Count number of zeroes in a binary number
17. Insert "1" after each repetition of "0" in binary number
18. Replace "abc" by "cc" within a word in alphabet {a,b,c}
19. Middle part of a word divided in 3 parts
20. Erase the second part of a word in alphabet {a,b,c}
21. Erase the first part of a word in alphabet {a,b,c}
22. Count the number of "a" written after "c"
23. Erase symbols symmetric regarding the middle of a word in alphabet {a,b,c}
24. Erase odd symbols of a word in alphabet {a,b,c}
25. Convert a binary number into a unary number
26. Convert a unary number into a binary number
27. Convert a binary number into an octal number
28. Convert an octal number into a binary number

II. Matrices, dynamic memory allocation

Write and debug a program, for instance in C language with Dev-C++ compiler

Digging gold loops – follow given lines and directions within a square matrix of an odd size; input matrix size, generate and print random matrix, and

- (a) print elements and their indices on a given loop
- (b) sort elements on a given loop



An example: 1,2,5

Input 5

matr:

```
36 16 85 47 21
63 27 76 65 54
23 98 23 52 84
64 36 97 56 28
34 87 23 96 45
```

(a)

```
a[0][0]=36
a[0][1]=16
a[0][2]=85
a[0][3]=47
a[0][4]=21
a[1][4]=54
a[2][4]=84
a[3][4]=28
a[4][4]=45
a[3][3]=56
a[2][2]=23
a[1][1]=27
```

(b)

matr:

```
16 21 23 27 28
63 85 76 65 36
23 98 84 52 45
64 36 97 56 47
34 87 23 96 54
```

Variants of the task:

- 1) 1,6,5
- 2) 1,2,7,4
- 3) 4,7,5,3
- 4) 5,8,3,4
- 5) 5,3,4
- 6) 5,8,3,4
- 7) 6,8,3,2
- 8) 6,2,3
- 9) 1,8,3,4
- 10) 6,5,3
- 11) 6,5,2
- 12) 6,8,3,2
- 13) 4,7,6
- 14) 1,6,4

For a greater variant use a reverse order of a variant $n \% 15 + 1$.

III. Two link lists and binary trees – an arbitrary insertion/deletion.

IV. Implement a hash table: insertion/deletion

V. Implement a 16-tree: insertion/deletion

VI. Implement algorithms considered but not implemented during lectures

