

Задание №2

Выполнил
студент гр. 5130904/30008
Ребдев Павел

С помощью нормальных алгоритмов Маркова преобразовать десятичное число в единичное.

Алфавит:

{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, | }

Правила:

1. $9 \rightarrow 8|$
2. $8 \rightarrow 7|$
3. $7 \rightarrow 6|$
4. $6 \rightarrow 5|$
5. $5 \rightarrow 4|$
6. $4 \rightarrow 3|$
7. $3 \rightarrow 2|$
8. $2 \rightarrow 1|$
9. $1 \rightarrow 0|$
10. $|9 \rightarrow 9|$
11. $|8 \rightarrow 8|$
12. $|7 \rightarrow 7|$
13. $|6 \rightarrow 6|$
14. $|5 \rightarrow 5|$
15. $|4 \rightarrow 4|$
16. $|3 \rightarrow 3|$
17. $|2 \rightarrow 2|$
18. $|1 \rightarrow 1|$
19. $|0 \rightarrow 0|$

20. 0 → «»

Ручной расчёт:

Исходные данные: 19

1. 0|9

2. 09|||||

3. 08|||||

4. 07|||||

5. 06|||||

6. 05|||||

7. 04|||||

8. 03|||||

9. 02|||||

10. 01|||||

11. 00|||||

12. 0|||||

13. |||||

Результат: |||||

Программа:

```
#include <iostream>
#include <string>
#include <list>
#include <utility>

int main()
{
    std::list< std::pair<std::string, std::string> > rules
    {
        {"9", "8|"},
        {"8", "7|"},
        {"7", "6|"},
        {"6", "5|"},
        {"5", "4|"},
        {"4", "3|"},
        {"3", "2|"},
        {"2", "1|"},
        {"1", "0|"},
    }
```

```

{"|9", "9|"},
{"|8", "8|"},
{"|7", "7|"},
{"|6", "6|"},
{"|5", "5|"},
{"|4", "4|"},
{"|3", "3|"},
{"|2", "2|"},
{"|1", "1|"},
{"|0", "0|"},
{"0", ""}
};
std::string input;
std::cout << "Write decimal number:\n\n";
std::cin >> input;
size_t steps = 1;
for (auto iter : rules)
{
    while (input.find(iter.first) != std::string::npos)
    {
        std::cout << steps++ << ". " << input << '\n';
        input.replace(input.find(iter.first), iter.first.size(), iter.second);
    }
}
std::cout << "\nResult: " << input << '\n';
std::cout << "Size: " << input.size() << '\n';
return 0;
}

```

Примеры работы программы:

Write decimal number:

```

19
1. 19
2. 18|
3. 17||
4. 16|||
5. 15||||
6. 14|||||
7. 13||||||
8. 12|||||||
9. 11|||||||
10. 0|1|||||||
11. 0|0|||||||
12. 00|||||||||
13. 0|||||||||

```

Result: |||||||||||
Size: 19

Write decimal number:

100

1. 100

2. 0|00

3. 00|||||||0

4. 00|||||||0|||||||

5. 00|||||||0|||||||

6. 00|||||||0|||||||

7. 00|||||||0|||||||

8. 00|||||0|||||||

9. 00|||||0|||||||

10. 00|||||0|||||||

11. 00|||||0|||||||

12. 00|0|||||||

13. 000|||||||

14. 00|||||||

15. 0|||||||

Result: |||

Size: 100

Write decimal number:

128

1. 128

2. 127|

3. 126||

4. 125|||

5. 124||||

6. 123|||||

7. 122||||||

8. 11|2|||||

9. 11|1|||||

10. 0|1|1|||||

11. 0|0|1|||||

12. 0|0|0|||||

13. 00|||||||0|||||||

14. 00|||||||0|||||||

15. 00|||||||0|||||||

16. 00|||||||0|||||||

17. 00|||||||0|||||||

18. 00|||||||0|||||||

19. 00|||||||0|||||||

20. 00|||||||0|||||||

21. 00|||||||0|||||||

22. 00|||||||0|||||||

23. 00|||||||0|||||||

24. 00|0|||||||

25. 000|||||||

26. 00|||||||

27. 0|||||||

Result: |||

Size: 128