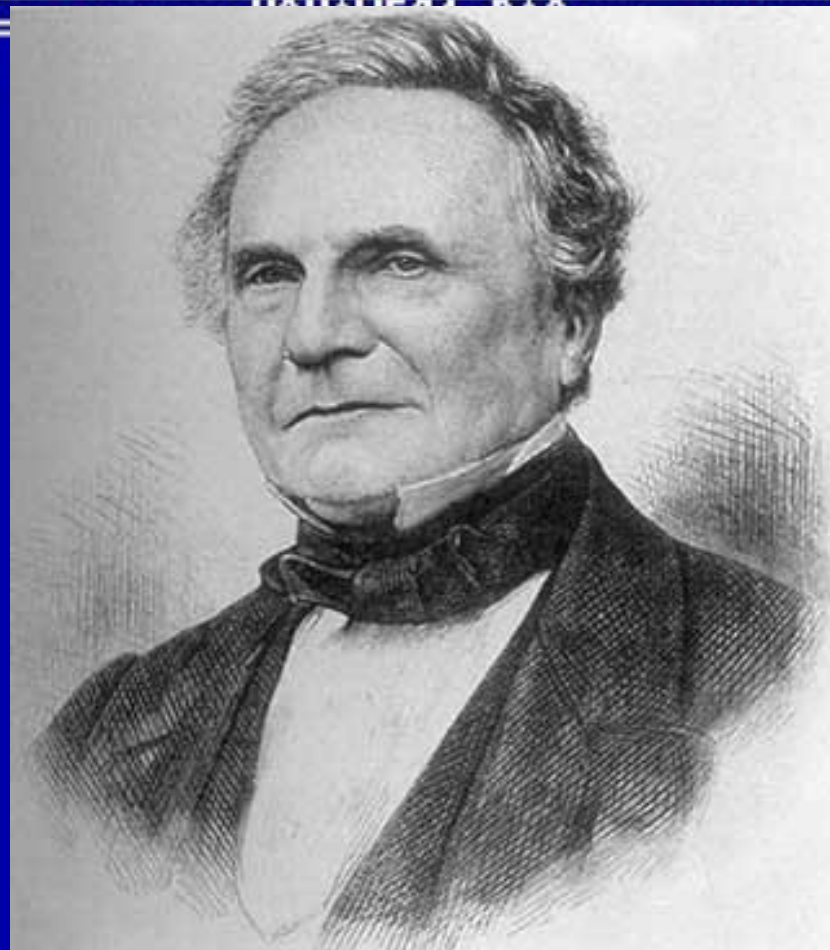


История развития языков программирования



Чарльз Бэббидж

1:1



File Edit Search Run Compile Debug Tools Options Window Help



2=[↑]



Джон Мокли

1:1



F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu



Марк-1

1:1

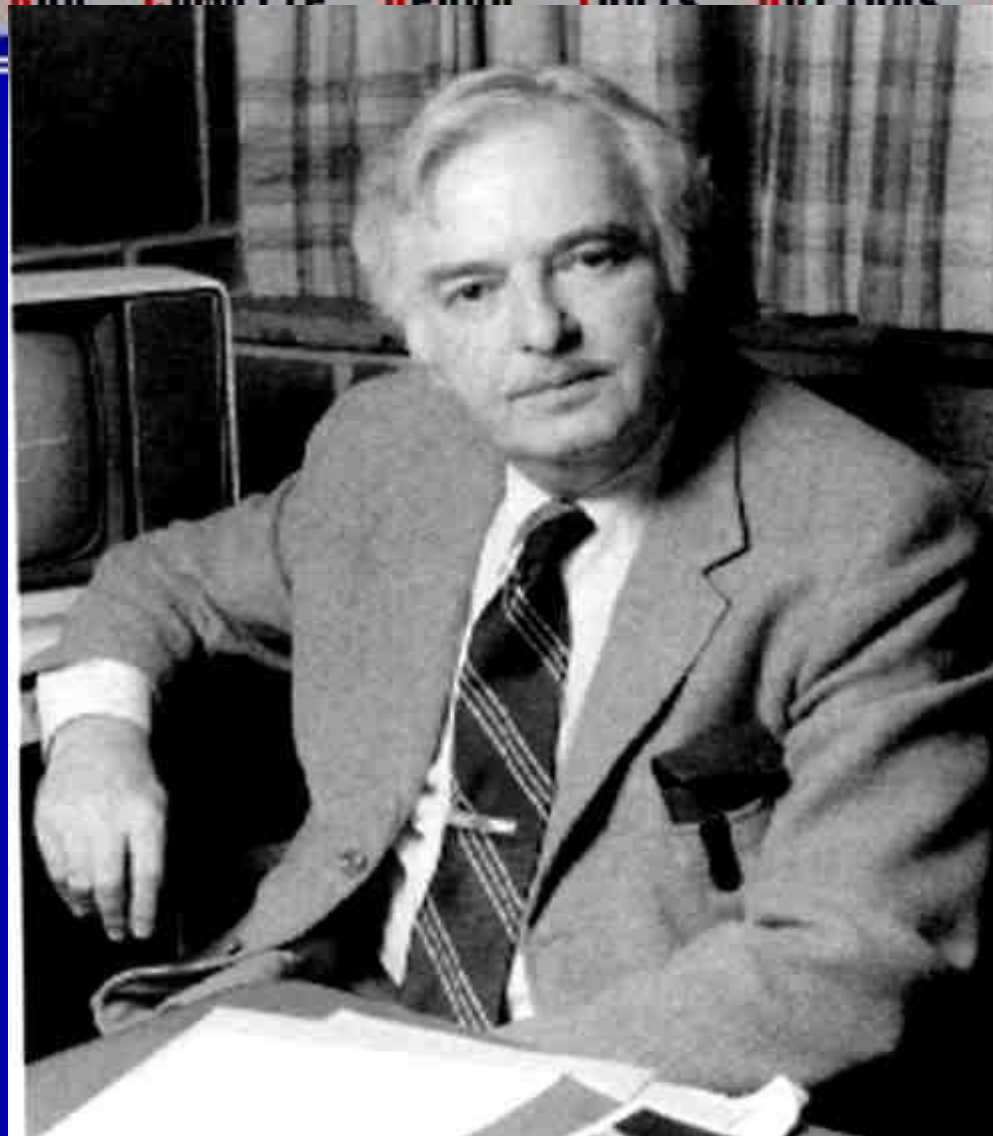
F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu

C FOR
CLASSIFICATION

STATEMENT NUMBER	1
---------------------	---

IDENTIFICATION

[illegible]



Джон Джордж Кемени

1:1

F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu

```
10 REM Диаграмма
20 SCREEN 9
30 COLOR 14, 1
40 DATA 11500, 2300, 9700, 5100, 12400, 8200
45 DATA "Токио", "Гамбург", "Москва", "Бангкок", "Мехико", "Париж"
50 LINE (40, 300)-(550, 300): LINE (40, 300)-(40, 20)
60 FOR k = 1 TO 6
70 READ m(k)
80 NEXT k
90 a = m(1)
100 FOR k = 2 TO 6
110 IF m(k) > a THEN a = m(k)
120 NEXT k
130 FOR k = 1 TO 6
140 READ name$(k)
150 NEXT k
160 FOR k = 1 TO 6
170 n = 10 * k + 3
```

Basic

← Immediate →



THE
C

PROGRAMMING
LANGUAGE



Algo168

File Edit Search Run Compile Debug Tools Options Window Help



NONAMED1.PAS

2=[↑]



Паскаль.Н.В

1:1



F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu

```

1 #include <font.h>
2 #include <stdio.h>
3 #include <stdlib.h>
4 #include <unistd.h>
5 #include <sys/types.h>
6 #include <sys/stat.h>
7
8 int main(int argc, char **argv)
9 {
10     int memory[32768], spointer;
11     int fd, len, buf, spointer, npointer;
12     int kl = 0;
13
14     if (argc <= 1)
15     {
16         printf("Usage: %s <brainfuck file>\n", argv[0]);
17         exit(1);
18     }
19
20     // Open brainfuck file
21     if ((fd = open(argv[1], O_RDONLY)) < 0) exit(1);
22
23     // Read the file byte by byte
24     len = spointer = 1;
25     while (len > 0)
26     {
27         len = read(fd, &buf, 1);
28         space[spointer] = buf;
29         spointer++;
30     }
31
32     close(fd);
33
34     for (npointer = 0; npointer < 32768; npointer++) memory[npointer] = 0;
35
36     len = spointer;
37     spointer = npointer = 0;
38
39     for (spointer = 0; spointer < len; spointer++)
40     {
41         switch(space[spointer])
42         {
43             // Increment pointer value
44             case '+':
45                 memory[npointer]++;
46                 break;
47             // Decrement pointer value
48             case '-':
49                 memory[npointer]--;
50                 break;
51             // Increment pointer
52             case '>':
53                 npointer++;
54                 break;
55             // Decrement pointer
56             case '<':
57                 npointer--;
58                 break;
59             // Print current pointer value
60             case '.':
61                 putchar(memory[npointer]);
62                 break;
63             // Read value and store in current pointer
64             case ',':
65                 memory[npointer] = getchar();
66                 break;
67             // Start loop
68             case '[':
69                 if (memory[npointer] == 0)
70                 {
71                     // Find matching ']'
72                     spointer++;
73                     while (kl > 0 || space[spointer] != ']')
74                     {
75                         if (space[spointer] == '[') kl++;
76                         if (space[spointer] == ',') kl--;
77                         // Go in right direction
78                         spointer++;
79                     }
80                 }
81                 break;
82             // End loop
83             case ']':
84                 if (memory[npointer] != 0)
85                 {
86                     // Find matching '['
87                     spointer--;
88                     while (kl < 0 || space[spointer] != '[')
89                     {
90                         if (space[spointer] == '[') kl--;
91                         if (space[spointer] == ',') kl++;
92                         // Go left
93                         spointer--;
94                     }
95                 }
96                 break;
97         }
98         putchar('\n');
99     }
100     return (0);
101 }

```



Visual Basic



Всем спасибо за внимание

1:1