

```

1  section .data
2  msg db 'Enter two digit Number:: ',0xa
3  msg_len equ $-msg
4  res db 10,'Multiplication of elements is:: '
5  res_len equ $-res
6  choice db 'Enter your Choice:',0xa
7  db '1.Successive Addition',0xa
8  db '2.Add and Shift method',0xa
9  db '3.Exit',0xa
10 choice_len equ $-choice
11 section .bss
12 num resb 03
13 num1 resb 01
14 result resb 04
15 cho resb 2
16 section .text
17 global _start
18 _start:
19 xor rax,rax
20 xor rbx,rbx
21 xor rcx,rcx
22 xor rdx,rdx
23 mov byte[result],0
24 mov byte[num],0
25 mov byte[num1],0
26 mov rax,1
27 mov rdi,1
28 mov rsi,choice
29 mov rdx,choice_len
30 syscall
31
32 mov rax,0
33 ;; read choice
34 mov rdi,0
35 mov rsi,cho
36 mov rdx,2
37 syscall
38 cmp byte[cho],31h
39 je a
40 cmp byte[cho],32h
41 je b
42 jmp exit
43 a:  call Succe_addition
44 jmp _start
45 b:  call Add_shift
46 jmp _start
47 exit:
48 mov rax,60
49 mov rdi,0
50 syscall
51 convert:
52 ;; comparing choice
53 ;; ASCII to Hex conversion

```

```
54 xor rbx,rbx
55 xor rcx,rcx
56 xor rax,rax
57 mov rcx,02
58 mov rsi,num
59 up1:
60 rol bl,04
61 mov al,[rsi]
62 cmp al,39h
63 jbe p1
64 sub al,07h
65 jmp p2
66 p1: sub al,30h
67 p2: add bl,al
68 inc rsi
69 loop up1
70 ret
71
72 display:
73 mov rcx,4
74 mov rdi,result
75 dup1:
76 rol bx,4
77 mov al,bl
78 and al,0fh
79 cmp al,09h
80 jbe p3
81 add al,07h
82 jmp p4
83 p3: add al,30h
84 p4:mov [rdi],al
85 inc rdi
86 loop dup1
87 mov rax,1
88 mov rdi,1
89 mov rsi,result
90 mov rdx,4
91 syscall
92 ret
93 Succe_addition:
94 mov rax,1
95 mov rdi,1
96 mov rsi,msg
97 mov rdx,msg_len
98 syscall
99 mov rax,0
100 mov rdi,0
101 mov rsi,num
102 mov rdx,3
103 syscall
104 call convert
105 mov [num1],bl
106 mov rax,1
107 mov rdi,1
108 mov rsi,msg
```

```
109 ;; Hex to ASCII conversion
110
111 mov rdx,msg_len
112 syscall
113 mov rax,0
114 mov rdi,0
115 mov rsi,num
116 mov rdx,3
117 syscall
118 call convert
119 xor rcx,rcx
120 xor rax,rax
121 mov rax,[num1]
122 repet:
123 add rcx,rax
124 dec bl
125 jnz repet
126 mov [result],rcx
127 mov rax,1
128 mov rdi,1
129 mov rsi,res
130 mov rdx,res_len
131 syscall
132 mov rbx,[result]
133 call display
134 ret
135 Add_shift:
136 mov rax,1
137 mov rdi,1
138 mov rsi,msg
139 mov rdx,msg_len
140 syscall
141 mov rax,0
142 mov rdi,0
143
144 mov rsi,num
145 mov rdx,3
146 syscall
147 call convert
148 mov [num1],bl
149 mov rax,1
150 mov rdi,1
151 mov rsi,msg
152 mov rdx,msg_len
153 syscall
154 mov rax,0
155 mov rdi,0
156 mov rsi,num
157 mov rdx,3
158 syscall
159 call convert
160 mov [num],bl
161 xor rbx,rbx
162 xor rcx,rcx
163 xor rdx,rdx
```

```
164 xor rax,rax
165 mov dl,08
166 mov al,[num1]
167 mov bl,[num]
168 p11:
169 shr bx,01
170 jnc p
171 add cx,ax
172 p:
173 shl ax,01
174 dec dl
175 jnz p11
176 mov [result],rcx
177 mov rax,1
178
179 mov rdi,1
180 mov rsi,res
181 mov rdx,res_len
182 syscall
183 ;dispmsg res,res_len
184 mov rbx,[result]
185 call display
186 ret
187
```