Capstone Engine is a framework for binary disassembly.
a powerful library that allows to disassemble binaries.
It is particularly useful if you want to automate some
of your reverse engineering analysis or identify known
pattern for evasion techniques.

```
# Output :        1000: cld
# Output :        1001: dec eax
# Output :        1002: and esp, 0xfffffff0
# Output :        1005: call 0x10ca
# Output :        100a: inc ecx
# Output :        100b: push ecx
# Output :        100c: inc ecx
# Output :        100d: push eax
# Output :        100e: push edx
# Output :        100f: push ecx
# Output :        1010: push esi
# Output :        1011: dec eax
# Output :        1012: xor edx, edx
# Output :        1014: dec eax
# Output :        1016: mov edx, dword ptr [edx + 0x60]
# Output :        1019: dec eax
# Output :        101a: mov edx, dword ptr [edx + 0x18]
# Output :        101d: dec eax
# Output :        101e: mov edx, dword ptr [edx + 0x20]
# Output :        1021: dec eax
# Output :        1022: mov esi, dword ptr [edx + 0x50]
# Output :        1025: dec eax
# Output :        1026: movzx ecx, word ptr [edx + 0x4a]
# Output :        102a: dec ebp
# Output :        102b: xor ecx, ecx
# Output :        102d: dec eax
# Output :        102e: xor eax, eax
# Output :        1030: lodsb al, byte ptr [esi]
# Output :        1031: cmp al, 0x61
# Output :        1033: jl 0x1037
# Output :        1035: sub al, 0x20
# Output :        1037: inc ecx
# Output :        1038: ror ecx, 0xd
# Output :        103b: inc ecx
# Output :        103c: add ecx, eax
# Output :        103e: loop 0x102d
# Output :        1040: push edx
# Output :        1041: inc ecx
# Output :        1042: push ecx
# Output :        1043: dec eax
# Output :        1044: mov edx, dword ptr [edx + 0x20]
# Output :        1047: mov eax, dword ptr [edx + 0x3c]
# Output :        104a: dec eax
# Output :        104b: add eax, edx
# Output :        104d: mov eax, dword ptr [eax + 0x88]
# Output :        1053: dec eax
```

Output :   1054: test eax, eax
Output :   1056: je 0x10bf
Output :   1058: dec eax
Output :   1059: add eax, edx
Output :   105b: push eax
Output :   105c: mov ecx, dword ptr [eax + 0x18]
Output :   105f: inc esp
Output :   1060: mov eax, dword ptr [eax + 0x20]
Output :   1063: dec ecx
Output :   1064: add eax, edx
Output :   1066: jecxz 0x10be
Output :   1068: dec eax
Output :   1069: dec ecx
Output :   106b: inc ecx
Output :   106c: mov esi, dword ptr [eax + ecx*4]
Output :   106f: dec eax
Output :   1070: add esi, edx
Output :   1072: dec ebp
Output :   1073: xor ecx, ecx
Output :   1075: dec eax
Output :   1076: xor eax, eax
Output :   1078: lodsb al, byte ptr [esi]
Output :   1079: inc ecx
Output :   107a: ror ecx, 0xd
Output :   107d: inc ecx
Output :   107e: add ecx, eax
Output :   1080: cmp al, ah
Output :   1082: jne 0x1075
Output :   1084: dec esp
Output :   1085: add ecx, dword ptr [esp + 8]
Output :   1089: inc ebp
Output :   108a: cmp ecx, edx
Output :   108c: jne 0x1066
Output :   108e: pop eax
Output :   108f: inc esp
Output :   1090: mov eax, dword ptr [eax + 0x24]
Output :   1093: dec ecx
Output :   1094: add eax, edx
Output :   1096: inc cx
Output :   1098: mov ecx, dword ptr [eax + ecx*2]
Output :   109b: inc esp
Output :   109c: mov eax, dword ptr [eax + 0x1c]
Output :   109f: dec ecx
Output :   10a0: add eax, edx
Output :   10a2: inc ecx
Output :   10a3: mov eax, dword ptr [eax + ecx*4]
Output :   10a6: dec eax
Output :   10a7: add eax, edx
Output :   10a9: inc ecx
Output :   10aa: pop eax
Output :   10ab: inc ecx
Output :   10ac: pop eax

Output :   10ad: pop esi

Output :   10ae: pop ecx

Output :   10af: pop edx

Output :   10b0: inc ecx

Output :   10b1: pop eax

Output :   10b2: inc ecx

Output :   10b3: pop ecx

Output :   10b4: inc ecx

Output :   10b5: pop edx

Output :   10b6: dec eax

Output :   10b7: sub esp, 0x20

Output :   10ba: inc ecx

Output :   10bb: push edx

Output :   10bc: jmp eax

Output :   10be: pop eax

Output :   10bf: inc ecx

Output :   10c0: pop ecx

Output :   10c1: pop edx

Output :   10c2: dec eax

Output :   10c3: mov edx, dword ptr [edx]

Output :   10c5: jmp 0x1021

Output :   10ca: pop ebp

Output :   10cb: dec ecx

Output :   10cc: mov esi, 0x5f327377

Output :   10d1: xor esi, dword ptr [edx]

Output :   10d3: add byte ptr [eax], al

Output :   10d5: inc ecx

Output :   10d6: push esi

Output :   10d7: dec ecx

Output :   10d8: mov esi, esp

Output :   10da: dec eax

Output :   10db: sub esp, 0x1a0

Output :   10e1: dec ecx

Output :   10e2: mov ebp, esp

Output :   10e4: dec ecx

Output :   10e5: mov esp, 0x29230002

Output :   10ea: or cl, byte ptr [ebx]

Output :   10ec: add al, 0x21

Output :   10ee: inc ecx

Output :   10ef: push esp

Output :   10f0: dec ecx

Output :   10f1: mov esp, esp

Output :   10f3: dec esp

Output :   10f4: mov ecx, esi

Output :   10f6: inc ecx

Output :   10f7: mov edx, 0x726774c

Output :   10fc: call ebp

Output :   10fe: dec esp

Output :   10ff: mov edx, ebp

Output :   1101: push 0x101

Output :   1106: pop ecx

Output :   1107: inc ecx

Output :   1108: mov edx, 0x6b8029

Output :   110d: call ebp

Output :   110f: push eax

Output :   1110: push eax

Output :   1111: dec ebp

Output :   1112: xor ecx, ecx

Output :   1114: dec ebp

Output :   1115: xor eax, eax

Output :   1117: dec eax

Output :   1118: inc eax

Output :   111a: dec eax

Output :   111b: mov edx, eax

Output :   111d: dec eax

Output :   111e: inc eax

Output :   1120: dec eax

Output :   1121: mov ecx, eax

Output :   1123: inc ecx

Output :   1124: mov edx, 0xe0df0fea

Output :   1129: call ebp

Output :   112b: dec eax

Output :   112c: mov edi, eax

Output :   112e: push 0x10

Output :   1130: inc ecx

Output :   1131: pop eax

Output :   1132: dec esp

Output :   1133: mov edx, esp

Output :   1135: dec eax

Output :   1136: mov ecx, edi

Output :   1138: inc ecx

Output :   1139: mov edx, 0x6174a599

Output :   113e: call ebp

Output :   1140: dec eax

Output :   1141: add esp, 0x240

Output :   1147: dec ecx

Output :   1148: mov eax, 0x646d63

Output :   114d: add byte ptr [eax], al

Output :   114f: add byte ptr [eax], al

Output :   1151: inc ecx

Output :   1152: push eax

Output :   1153: inc ecx

Output :   1154: push eax

Output :   1155: dec eax

Output :   1156: mov edx, esp

Output :   1158: push edi

Output :   1159: push edi

Output :   115a: push edi

Output :   115b: dec ebp

Output :   115c: xor eax, eax

Output :   115e: push 0xd

Output :   1160: pop ecx

Output :   1161: inc ecx

Output :   1162: push eax

Output :   1163: loop 0x1161

Output :   1165: mov word ptr [esp + 0x54], 0x101

Output :   116c: dec eax

Output :   116d: lea eax, [esp + 0x18]

Output :   1171: mov byte ptr [eax], 0x68

Output :   1174: dec eax

Output :   1175: mov esi, esp

Output :   1177: push esi

Output :   1178: push eax

Output :   1179: inc ecx

Output :   117a: push eax

Output :   117b: inc ecx

Output :   117c: push eax

Output :   117d: inc ecx

Output :   117e: push eax

Output :   117f: dec ecx

Output :   1180: inc eax

Output :   1182: inc ecx

Output :   1183: push eax

Output :   1184: dec ecx

Output :   1185: dec eax

Output :   1187: dec ebp

Output :   1188: mov ecx, eax

Output :   118a: dec esp

Output :   118b: mov ecx, eax

Output :   118d: inc ecx

Output :   118e: mov edx, 0x863fcc79

Output :   1193: call ebp

Output :   1195: dec eax

Output :   1196: xor edx, edx

Output :   1198: dec eax

Output :   1199: dec edx

Output :   119b: mov ecx, dword ptr [esi]

Output :   119d: inc ecx

Output :   119e: mov edx, 0x601d8708

Output :   11a3: call ebp

Output :   11a5: mov ebx, 0x56a2b5f0

Output :   11aa: inc ecx

Output :   11ab: mov edx, 0x9dbd95a6

Output :   11b0: call ebp

Output :   11b2: dec eax

Output :   11b3: add esp, 0x28

Output :   11b6: cmp al, 6

Output :   11b8: jl 0x11c4

Output :   11ba: cmp bl, 0xe0

Output :   11bd: jne 0x11c4

Output :   11bf: mov ebx, 0x6f721347

Output :   11c4: push 0

Output :   11c6: pop ecx

Output :   11c7: inc ecx

Output :   11c8: mov edx, ebx

Output :   11ca: call ebp

Output :   11cc: add byte ptr [eax], al

Output :   11ce: add byte ptr [eax], al

Output :   11d0: clc

Output :   11d1: inc ecx

Output :   11d2: add byte ptr [eax], al

Output :   11d4: add byte ptr [eax], al

Output :   11d6: add byte ptr [eax], al