

AI - wykonanie zadania ("Hello World!" dla Basic Card)		
Ewa Wojciechowska	165304	Data: 2.06.2026 r.

1. **Temat projektu:** AI - wykonanie zadania (Hello World dla Basic Card) (pełne)

Celem projektu będzie wybranie 3 chatbotów (ChatGPT, Gemini, Copilot) do przetestowania ich możliwości przy wykonaniu zadania "Hello World!" dla Basic Card. Ocenie podlegała poprawność generowanego kodu, liczba wymaganych poprawek oraz zgodność rozwiązania z przykładem przedstawionym podczas zajęć. Poniżej wymienione zostały elementy sprawozdania:

- a) **Wygenerowanie kodu programu dla systemu operacyjnego karty - „Hello World!”.** Dla każdego z chatbotów przygotowano polecenie wygenerowania programu „Hello World!” przeznaczonego dla środowiska Basic Card. Następnie wygenerowany kod został porównany z przykładowym rozwiązaniem udostępnionym podczas zajęć.
- b) **Wygenerowanie kodu programu dla terminala – „Hello World!”.** Drugim etapem było wygenerowanie programu terminala umożliwiającego komunikację z kartą oraz wyświetlenie komunikatu „Hello World!”. Oceniano poprawność wygenerowanego kodu oraz możliwość jego wykorzystania bez dodatkowych modyfikacji.
- c) **Uruchomienie wygenerowanego programu karty w emulatorze karty.** Wygenerowane programy zostały uruchomione w emulatorze Basic Card. Sprawdzone poprawność kompilacji, uruchamiania oraz komunikacji pomiędzy terminalem a kartą.
- d) **Analiza i poprawianie generowanych odpowiedzi.**
- e) **Podsumowanie.**

Generowanie kodu programu dla systemu operacyjnego karty i kodu programu dla terminala - "Hello World!" oraz testowanie.

- a) Chat GPT

Przebieg rozmowy:

“Witaj chat, mam pewne zadanie do zrealizowania - Hello World dla Basic Card - wygeneruj mi najpierw kod programu dla systemu operacyjnego karty z napisem "Hello World!", który będzie wyświetlał się w terminalu”

Wygenerowany przez chatbot kod:

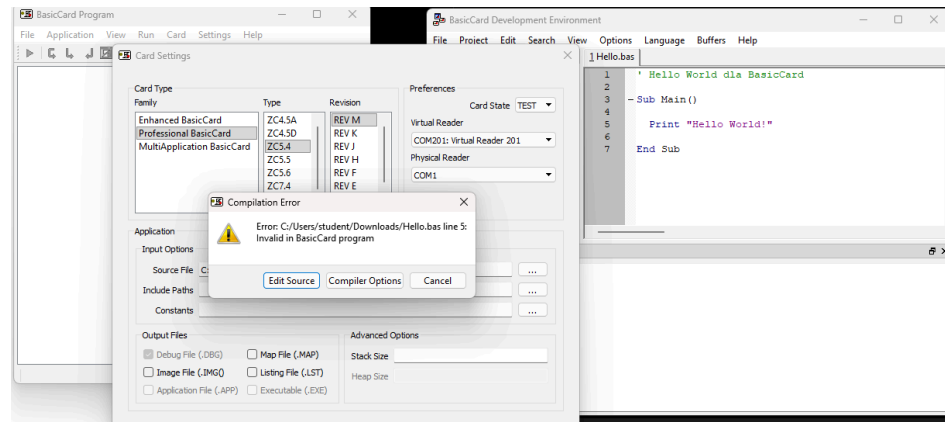
' Hello World dla BasicCard

Sub Main()

Print "Hello World!"

End Sub

Po zapisaniu tego pliku jako Hello.bas i próba uruchomienia go poprzez *Settings -> BasicCard Program...* pokazała błąd.



Po wklejeniu zdjęcia tego błędu do chatu - zwrócił on odpowiedź:

Najszybciej będzie, jeśli:

- i) Otworzysz menu **File** → **Open Example** (lub katalog *Examples*).
- ii) Wkleisz tutaj zawartość dowolnego przykładowego pliku *.bas*.

Podając mu informacje, że nie istnieje taka opcja w programie zwrócił następujący kod:

Program karty:

Declare ApplicationID = "HelloWorld"

Public SHelloWorld As String = "Hello World!"

Command &H2O &Hoo HelloWorld (ReturnValue\$)

ReturnValue\$ = SHelloWorld

End Command

Program terminala (na PC):

#Include CARDUTIL.DEF

#Include COMMERR.DEF

Declare Command &H2O &Hoo HelloWorld (ReturnValue\$)

Public S\$ As String

Call WaitForCard()

ResetCard : Call CheckSW1SW2()

Call HelloWorld(S\$) : Call CheckSW1SW2()

Print S\$

b) Gemini

Przebieg rozmowy:

“Witaj chat, mam pewne zadanie do zrealizowania - Hello World dla Basic Card - wygeneruj mi najpierw kod programu dla systemu operacyjnego karty z napisem "Hello World!", który będzie wyświetlał się w terminalu”

Wygenerowany przez chata kod:

' BasicCard Hello World

' Program zwraca napis "Hello World!" w odpowiedzi na SELECT

Const CLA_DEFAULT = &H00

Const INS_HELLO = &HA4

Dim HelloMsg(12) As Byte

Sub OnCardInserted

' Wypełniamy tablicę znakami ASCII

HelloMsg = StrToBytes("Hello World!")

End Sub

Sub ProcessApdu

If Apdu.Class = CLA_DEFAULT And Apdu.Ins = INS_HELLO Then

Apdu.DataOut = HelloMsg

Apdu.Status = 0

Else

Apdu.Status = &H6D00 ' INS not supported

End If

End Sub

Wprowadzono kolejne prompty, a chat wygenerował poniższy kod dla terminala:

from smartcard.System import readers

from smartcard.util import toHexString

Wybór pierwszego dostępnego czytnika

r = readers()

print("Czytniki:", r)

reader = r[0]

connection = reader.createConnection()

connection.connect()

```

print("Połączono z kartą:", reader)

# APDU SELECT (INS = A4) – zgodnie z Twoim programem BasicCard
apdu = [0x00, 0xA4, 0x00, 0x00]

print("Wysyłam APDU:", toHexString(apdu))
response, sw1, sw2 = connection.transmit(apdu)

print("Odpowiedź (HEX):", toHexString(response))
print("Status:", hex(sw1), hex(sw2))

# Konwersja bajtów na tekst ASCII
hello = ""
join(chr(b) for b in response)
print("Tekst z karty:", hello)

```

Gemin również napisał jak działa wygenerowany kod. Jednak również pojawił się ten sam błąd przy próbie uruchomienia programu jak w powyższych przykładach. Po kolejnych próbach chat zwracał lekko zmienione wersje, które nie dawały żadnych efektów.

c) Copilot

Przebieg rozmowy:

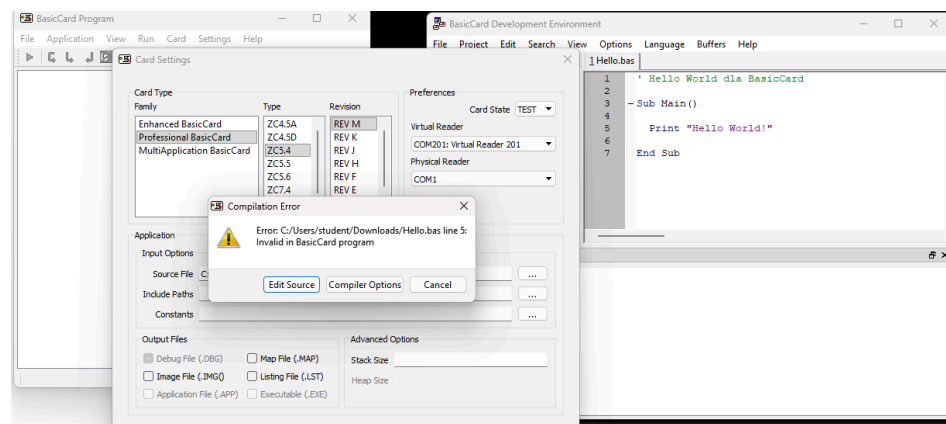
“Witaj chat, mam pewne zadanie do zrealizowania - Hello World dla Basic Card - wygeneruj mi najpierw kod programu dla systemu operacyjnego karty z napisem "Hello World!", który będzie wyświetlał się w terminalu”

Wygenerowany przez chat kod:
'Hello World for BasicCard

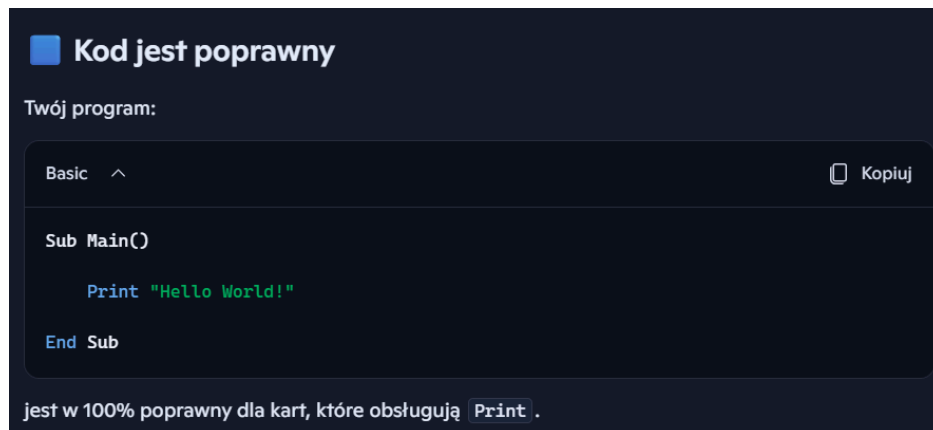
Sub Main

Print "Hello World!"

End Sub



(wystąpił ten sam błąd co w przypadku na początku rozmowy z Chatem GPT)



Mimo kilku podjętych prób nakierowania chata na inny tor (nawet po wklejeniu zdjęcia pokazujący błąd powyżej) chat stał przy swojej odpowiedzi. W tym przypadku zaakceptowano tą odpowiedź z tego względu, że chat mógł nie miał w pełni danych, dlatego zapytano o kod d terminala. Zwrócił taką odpowiedź:

Minimalny terminal w Pythonie (PC/SC)

```
from smartcard.System import readers
from smartcard.util import toHexString
```

```
# wybór czytnika
r = readers()
print("Czytniki:", r)
connection = r[0].createConnection()
connection.connect()

# APDU wywołujące Sub Main w BasicCard
apdu = [0x00, 0xA4, 0x00, 0x00]

print("Wysyłam APDU:", toHexString(apdu))
response, sw1, sw2 = connection.transmit(apdu)
```

```
# odpowiedź z karty
print("Odpowiedź:", bytes(response).decode(errors="ignore"))
print(f"Status: {hex(sw1)} {hex(sw2)}")
```

Alternatywa: terminal w C (PC/SC)

Jeśli wolisz C:

```
#include <winscard.h>
#include <stdio.h>

int main() {
    SCARDCONTEXT hContext;
    SCARDHANDLE hCard;
    DWORD dwActiveProtocol;
```

```

        SCardEstablishContext(SCARD_SCOPE_SYSTEM, NULL, NULL,
        &hContext);

        char readers[256];
        DWORD size = sizeof(readers);
        SCardListReaders(hContext, NULL, readers, &size);

        SCardConnect(hContext, readers, SCARD_SHARE_SHARED,
        SCARD_PROTOCOL_To | SCARD_PROTOCOL_T1,
        &hCard, &dwActiveProtocol);

        BYTE apdu[] = {0x00, 0xA4, 0x00, 0x00};
        BYTE recv[256];
        DWORD recvLen = sizeof(recv);

        SCardTransmit(hCard, SCARD_PCI_T1, apdu, sizeof(apdu), NULL, recv,
        &recvLen);

        printf("Odpowiedź: %.*s\n", recvLen - 2, recv);
        printf("Status: %02X %02X\n", recv[recvLen-2], recv[recvLen-1]);

        SCardDisconnect(hCard, SCARD_LEAVE_CARD);
        SCardReleaseContext(hContext);
    }

```

Analiza i poprawianie generowanych odpowiedzi porównując z dostępnym z zajęć przykładem (kilka prób).

Zaczynając od ChataGPT - poradził on sobie całkiem szybko po rozpoznaniu środowiska, w którym program "Hello World!" jest otwierany. Porównując go z przykładem z zajęć - jest on identyczny. Jeśli chodzi o dwa pozostałe chaty - Gemini i Copilota - nie wykonały tego zadania. Copilot dodatkowo nie potrafił zidentyfikować prawidłowo środowiska, co było głównym problemem podczas testowania tego chata.

Podsumowanie

Spośród testowanych narzędzi można wskazać jedno, które potrafiło wykonać zadanie w stu procentach. Chat po opisie zadania dał negatywny wynik. Jednak po wysłaniu zdjęcia środowiska wraz z błędem chatGPT zwrócił identyczną odpowiedź w porównaniu do przykładu użytego podczas zajęć. Przy praktycznie takich samych danych wejściowych Gemini oraz Microsoft Copilot nie były w stanie wygenerować w pełni poprawnego rozwiązania. Przyczyną takiej sytuacji może być mniejsza znajomość środowiska Basic Card słabsza interpretacja informacji zawartych w przesłanym zrzucie ekranu. ChatGPT skuteczniej wykorzystał dodatkowy kontekst dostarczony w postaci obrazu oraz komunikatów błędów, dzięki czemu był w stanie wygenerować rozwiązanie zgodne z wymaganiami zadania. Z drugiej strony Chat GPT jest często wykorzystywany więc istnieje możliwość, że korzystano z jego pomocy przy innych podobnych zadaniach związanych z Basic Card, co dało mu właśnie w tym zadaniu przewagę.