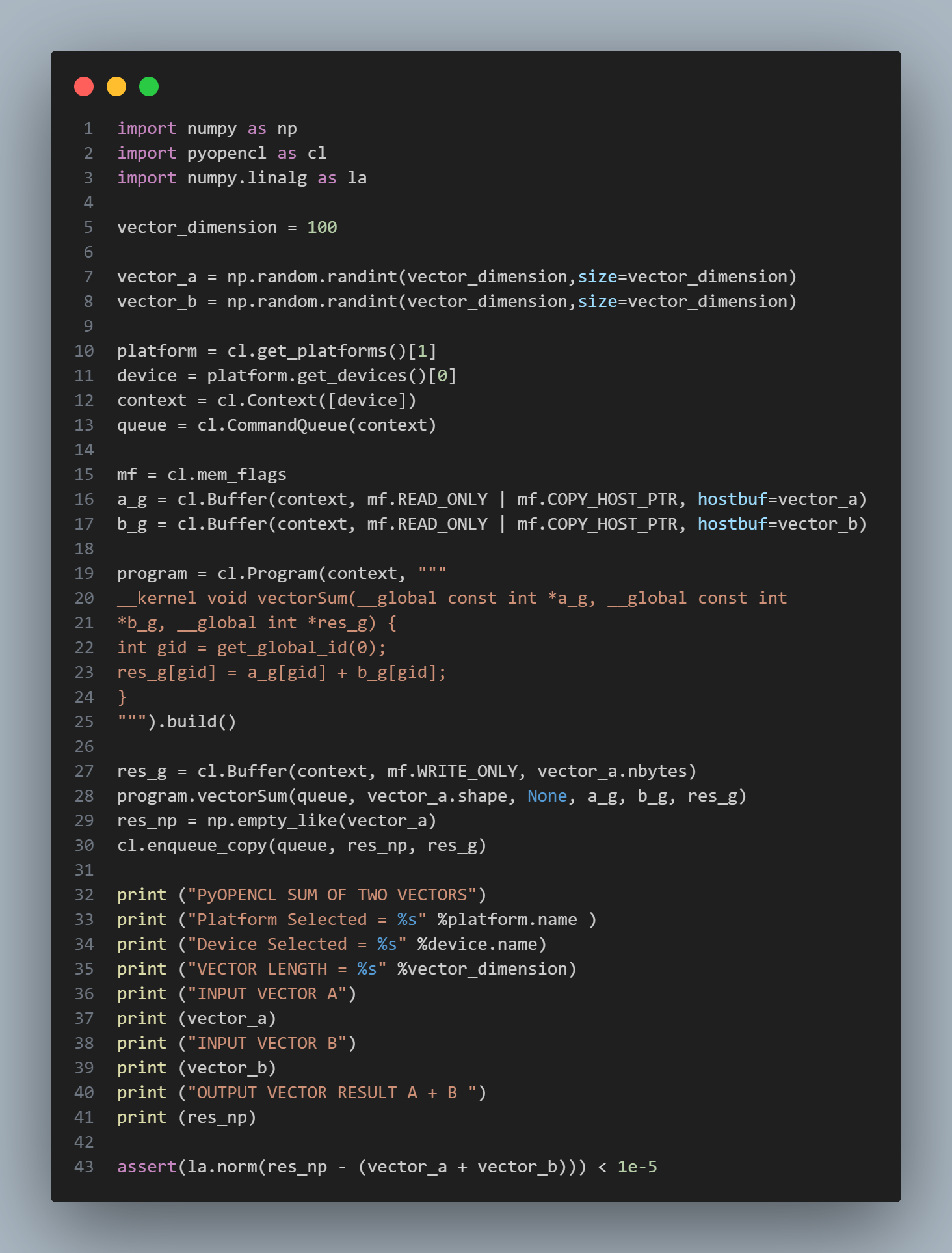
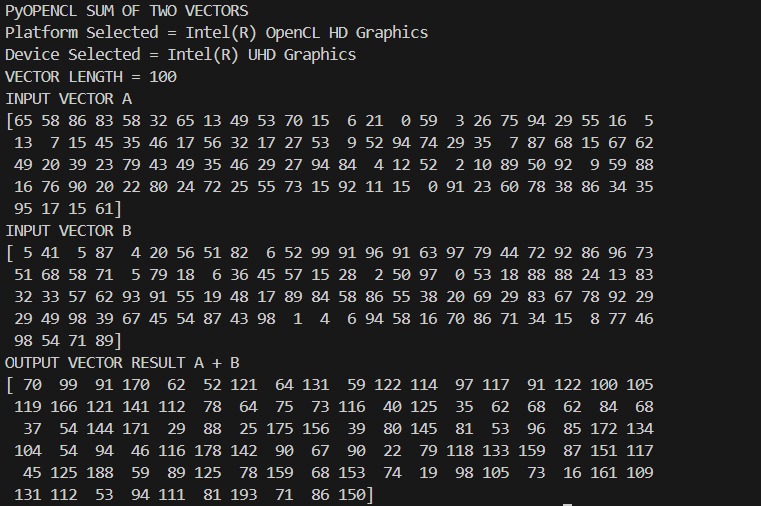
Nama: Ridwan Akrom, Prawinrangga Caesar NPB.

**Building Applications with PyOpenCL**

**Program**



**Hasil Program**



**Analisa**

Program ini merupakan implementasi operasi penjumlahan dua vektor menggunakan PyOpenCL, sebuah library Python yang mendukung komputasi paralel berbasis OpenCL. Program dimulai dengan menghasilkan dua vektor, vector\_a dan vector\_b, yang masing-masing memiliki panjang 100 elemen. Elemen-elemen ini merupakan bilangan acak dalam rentang 0 hingga 99. Vektor tersebut nantinya akan digunakan sebagai data masukan untuk komputasi di perangkat GPU.

Selanjutnya, program memilih platform OpenCL kedua (cl.get\_platforms()[1]) dan perangkat pertama dari platform tersebut (platform.get\_devices()[0]). Dalam hasil eksekusi program, perangkat yang dipilih adalah Intel UHD Graphics. Konteks (context) dibuat untuk menghubungkan aplikasi dengan perangkat GPU yang dipilih, sementara antrean perintah (queue) digunakan untuk mengirimkan instruksi dari host (CPU) ke perangkat.

Setelah itu, program mengalokasikan buffer memori untuk menyimpan data input (a\_g dan b\_g) dan output (res\_g). Buffer a\_g dan b\_g hanya bisa dibaca oleh kernel GPU, sedangkan buffer res\_g hanya bisa ditulis oleh kernel. Data vektor vector\_a dan vector\_b disalin ke buffer masukan menggunakan flag COPY\_HOST\_PTR.

Kernel OpenCL didefinisikan dalam kode sebagai fungsi vectorSum, yang menerima tiga parameter: dua buffer input (a\_g dan b\_g) dan satu buffer output (res\_g). Kernel ini menjumlahkan elemen-elemen dari dua vektor berdasarkan indeks global (get\_global\_id(0)) dan menyimpan hasilnya di buffer output. Kernel ini kemudian dibangun menggunakan konteks GPU yang telah dibuat.

Setelah kernel dijalankan, hasil komputasi pada buffer res\_g disalin kembali ke host (CPU) dalam bentuk array NumPy res\_np. Program kemudian membandingkan hasil GPU (res\_np) dengan hasil perhitungan CPU (vector\_a + vector\_b) menggunakan norma vektor. Jika selisihnya kecil (kurang dari 1e-5), hasil dianggap valid.

Sebagai hasil akhir, program mencetak informasi platform dan perangkat yang digunakan, nilai vektor input (vector\_a dan vector\_b), serta hasil penjumlahan kedua vektor (res\_np). Program ini menunjukkan keunggulan komputasi paralel menggunakan OpenCL, terutama untuk dataset besar. Namun, untuk dataset kecil seperti panjang vektor 100, overhead transfer data antara CPU dan GPU dapat mengurangi efisiensi. Untuk peningkatan, program dapat dibuat lebih dinamis dalam pemilihan platform dan memanfaatkan ukuran data yang lebih besar.