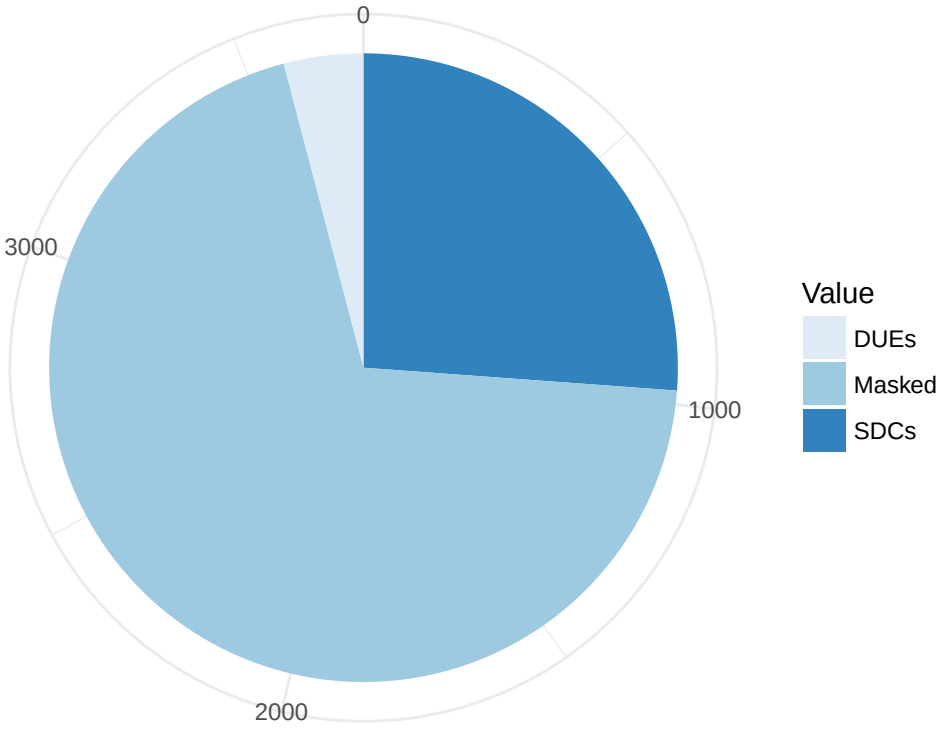
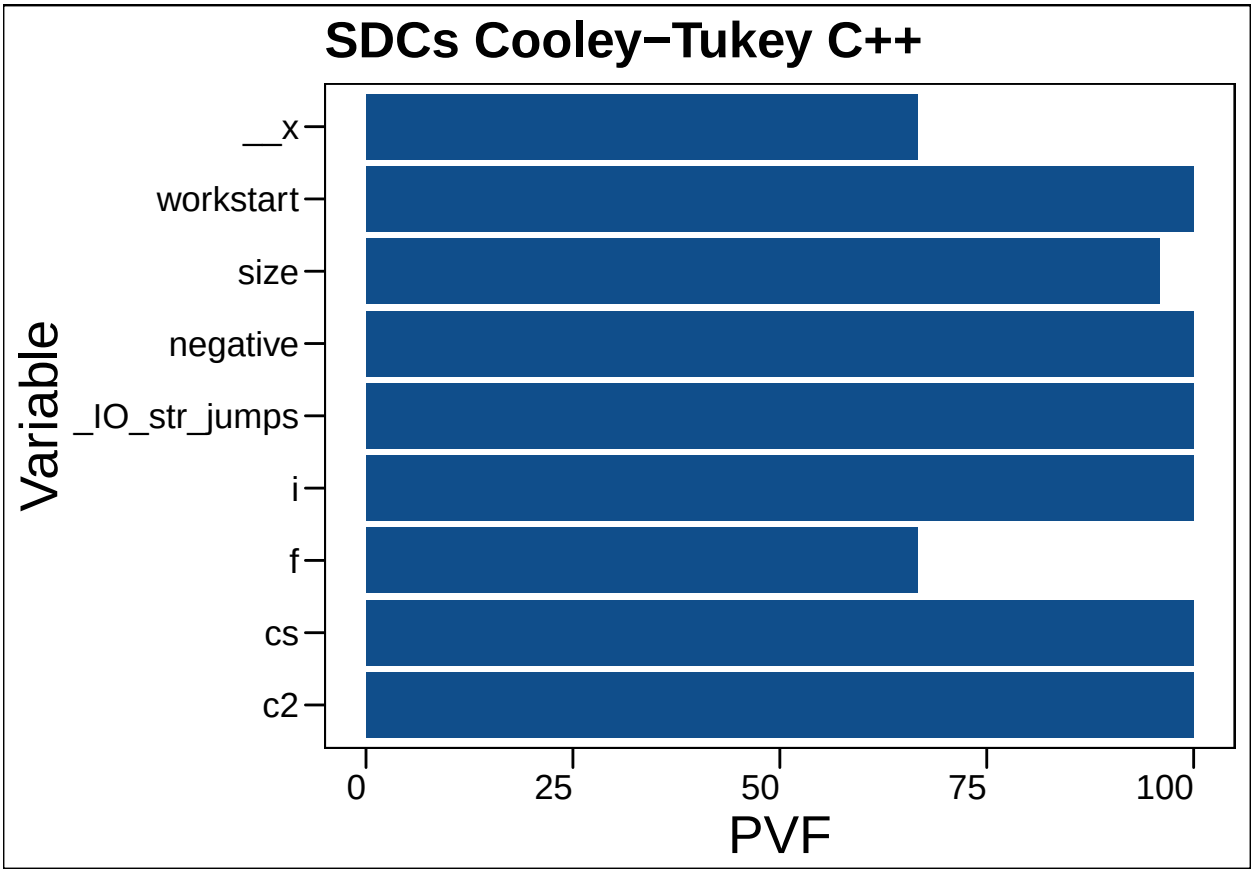
Análise das injeções

Análise Cooley-Tukey C++

Na implementação em C++, foram injetados 3723 falhas. No gráfico a seguir, é possível ver a distribuição dos resultados das injeções.

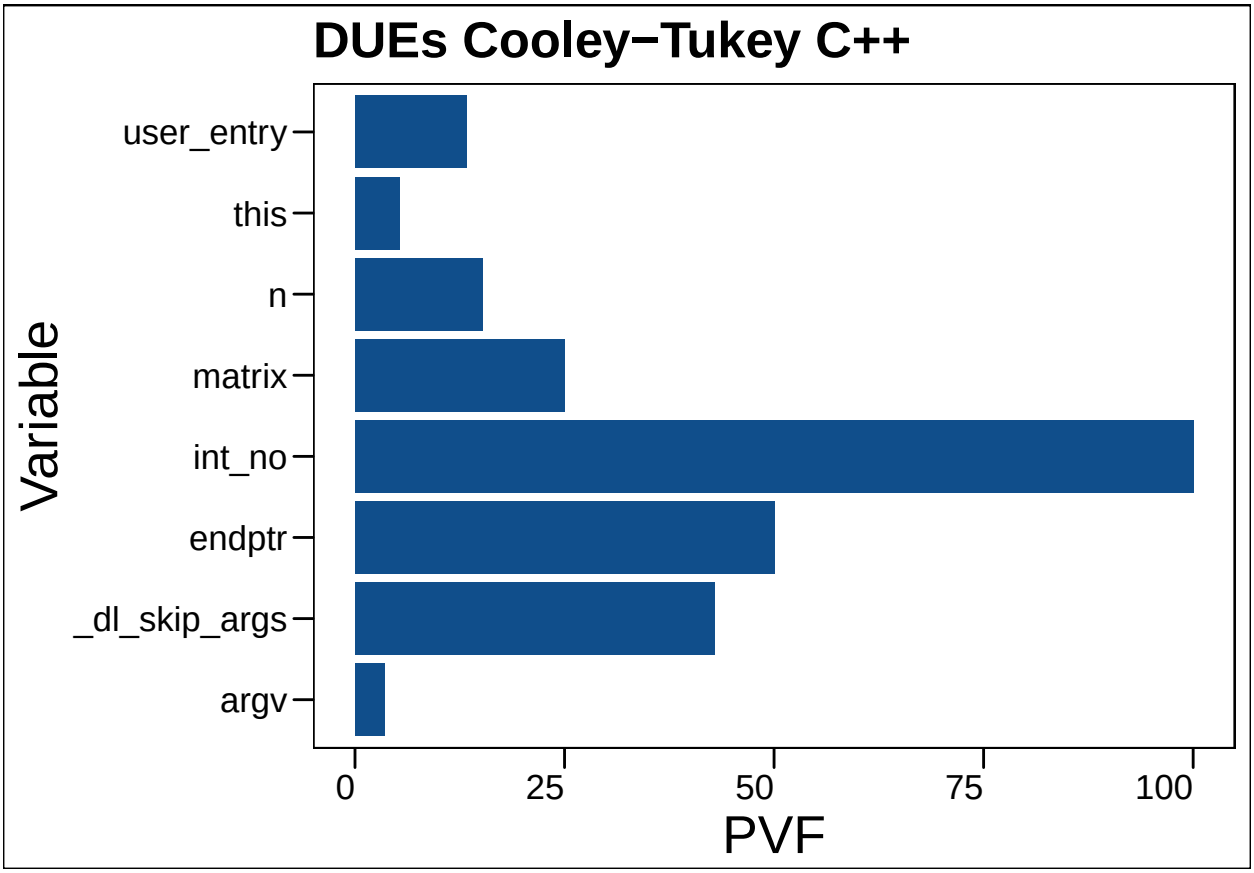


De acordo com os dados obtidos, cerca de 25% das falhas resultaram em SDCs, enquanto os DUEs representam menos de 5%. Felizmente, a grande maioria das injeções não resultou em alterações na saída.



Verificando as 10 variáveis mais vulneráveis, percebe-se que apenas duas entre elas fazem parte da aplicação em si, sendo as outras vindas das bibliotecas utilizadas. Com isso, concluimos que, para que a aplicação seja protegida da melhor maneira possível, é preferível que não sejam utilizadas bibliotecas (ou sejam utilizadas o mínimo possível). No caso deste trabalho, o foco ficará nas variáveis que podem ser protegidas, ou seja, as criadas pelos alunos.

Nesta análise, temos as váriaveis *i* e *size* como mais vulneráveis em questão de SDCs. Como tratam-se de variáveis de controle, uma simples duplicação (ou triplicação) não deve resultar em um *overhead* significativo e, portanto, provavelmente será uma boa solução na etapa seguinte do trabalho.



O mesmo se repete se tratando de DUEs; as variáveis que mais os causam são as externas. No entanto, também temos a variável n que, assim como as abordadas anteriormente, é uma variável de controle facilmente duplicável, e a variável *matrix*, responsável pela estrutura da aplicação. Para proteger esta, serão utilizadas técnicas mais sofisticadas que uma simples duplicação.