

DA SUBREGIONAL AO PÓDIO: PERFIS DE EDITORES EM DOIS CICLOS DA MARATONA SBC

Bruno Ribas, UnB

Resumo

Este é um artigo informativo e descritivo: a partir da telemetria de uso de editores em seis competições da Maratona SBC (dois ciclos, três fases, mais de 3.000 times), ele sumariza um achado central. O Visual Studio Code domina em todas as fases (73% a 83% das equipes), mas entre os 30 melhores os IDEs robustos praticamente somem e o Vim sobe para 23% a 37%, padrão que chamamos de "vale dos IDEs". A análise por país ainda revela perfis culturais nítidos, com destaque para o Brasil como o país do Vim. Não buscamos relação causal entre ferramenta e desempenho.

1. Dados e método

A Direção de Sistemas mantém em cada máquina das sedes uma coleta automática de telemetria simples: tempo agregado em que cada editor ficou aberto e um instantâneo de CPU e RAM. Essa coleta roda sobre a infraestrutura padronizada das sedes: o sistema de julgamento BOCA [1] e o ambiente Maratona Linux [2, 3], que já traz pré-instalados todos os editores aqui analisados. Os placares oficiais são públicos (<https://scorelatam.naquadah.com.br/>).

Analisamos dois ciclos completos, 2024–25 e 2025–26, cada um com as três fases da competição: a Primeira Fase (a regional brasileira), a Final Latam e o PDA. As três fases de um mesmo ciclo não são amostras independentes, e sim um funil: os times que disputam a Final já passaram por uma Primeira Fase, e os que chegam ao PDA vieram da Final. A unidade que se repete é o ciclo: um achado vale mais quando reaparece nos dois, não por ser contado seis vezes. Em números, são 855, 495 e 40 times no ciclo 24–25 e 1.047, 578 e 52 no ciclo 25–26 (Primeira Fase, Final e PDA, nessa ordem), somando mais de 3.000 registros, um por time em cada evento.

A cobertura varia por fase. Na Primeira Fase, ainda que alguns países usem nossa regional, nem todos adotam o sistema centralizado, de modo que os dados de editores são essencialmente dos times brasileiros. Na Final Latam, distribuída em uma sede por país latino-americano (a brasileira é a Final Brasileira), todas rodam o Maratona Linux, garantindo cobertura completa. O PDA, em sede única, também oferece cobertura total: 40 times oficiais em 2025 e 41 em 2026, mais 11 equipes convidadas (CCL) em 2026. Assim, na Tabela 1 a coluna Reg é essencialmente brasileira, enquanto Fin e PDA abrangem toda a América Latina; é também o que torna confiável a análise por país.

Adotamos o critério dos relatórios internos: um editor é considerado **utilizado** por um time se permaneceu aberto por ao menos 60 minutos durante a prova, que dura 300 minutos. Um mesmo time pode usar mais de um editor.

2. Domínio de VS Code e crescimento do Vim

TABELA 1: ADOÇÃO DE EDITORES (% DE TIMES COM ≥ 60 MIN ABERTO) EM SEIS COMPETIÇÕES.

Editor	Ciclo 24–25			Ciclo 25–26		
	Reg	Fin	PDA	Reg	Fin	PDA
N (times)	855	495	40	1047	578	52
Visual Studio Code	74,9	73,5	82,5	82,0	79,6	80,8
gedit	23,7	27,5	37,5	24,3	23,4	5,8
Vim	5,4	6,5	27,5	5,3	6,1	19,2
PyCharm	13,7	7,7	2,5	12,4	10,9	3,8
CLion	8,8	12,1	5,0	12,5	16,4	1,9
IntelliJ IDEA	8,2	3,8	0,0	6,0	5,2	0,0
Code::Blocks	16,5	20,6	7,5	11,6	15,1	0,0

A Tabela 1 apresenta o panorama. O Visual Studio Code [4] é o editor mais usado em todas as competições, com adoção entre 73% e 83%, praticamente estável entre fases e ciclos. O gedit [5] oscila e é sensível à imagem do PDA: cai de 37,5% em 2025 para 5,8% em 2026, indicando que sua presença como *default* pesa na adoção; é prudente lê-lo como medida de **disponibilidade**, não de **preferência**.

O Vim [6] mostra um padrão mais informativo. Na Regional e na Final Latam sua adoção fica entre 5% e 7%, mas no PDA salta para 27,5% em 2025 e 19,2% em 2026, aumento de 3 a 5 vezes em ambos os ciclos. Como o Vim raramente é aberto por acidente, a leitura mais plausível é que o salto reflete a maior densidade de competidores experientes em sede única.

3. O "vale dos IDEs" entre os Top-30

TABELA 2: ADOÇÃO DE EDITORES (%) ENTRE AS 30 MELHORES EQUIPES DO PLACAR OFICIAL DE CADA EVENTO.

Editor	Ciclo 24–25			Ciclo 25–26		
	Reg	Fin	PDA	Reg	Fin	PDA
Visual Studio Code	56,7	80,0	83,3	73,3	76,7	76,7
gedit	16,7	23,3	43,3	30,0	30,0	6,7
Vim	23,3	36,7	33,3	26,7	23,3	23,3
PyCharm	0,0	0,0	0,0	0,0	0,0	0,0
CLion	3,3	0,0	3,3	10,0	3,3	3,3
IntelliJ IDEA	0,0	0,0	0,0	0,0	0,0	0,0
Code::Blocks	0,0	3,3	10,0	3,3	3,3	0,0

A Tabela 2 restringe a análise às 30 melhores equipes. Dois pontos se sobressaem. Primeiro, **IntelliJ IDEA e PyCharm [7] não aparecem no Top-30 de nenhum dos seis eventos**; CLion e Code::Blocks aparecem, mas com adoção máxima de 10% e tipicamente abaixo de 4%. A fração de equipes que usaram qualquer IntelliJ ou Code::Blocks, no conjunto completo, fica entre 35% e 37% nas Regionais e Finais, e despenca para 5,8% a 12,5% no PDA. Esse é o "vale dos IDEs", padrão que reaparece nas três fases dos dois ciclos.

Segundo, o Vim ocupa o espaço deixado pelos IDEs na topo: em todas as seis competições, sua adoção no Top-30 fica entre 23% e 37%, faixa estável que não se vê quando olhamos todos os times. Editores leves (gedit, Vim, Geany) são, portanto, sistematicamente sobre-representados entre os melhores; IDEs pesados são sub-representados. Não afirmamos causa: pode ser que competidores que treinam mais prefiram editores ágeis, e não que o editor os faça melhores.

4. Perfis culturais por país

TABELA 3: ADOÇÃO DE EDITORES (%) POR PAÍS, NA FINAL LATAM DE 2025 (FOCO EM PAÍSES COM ≥ 20 EQUIPES). VSC = VISUAL STUDIO CODE; PYC = PYCHARM; CLI = CLION; C::B = CODE::BLOCKS; GN = GEANY.

País (n)	VSC	gedit	Vim	CLi	PyC	C::B	Gn
México (133)	82,0	25,6	4,5	24,1	10,5	9,0	3,0
Bolívia (76)	77,6	31,6	0,0	14,5	7,9	42,1	6,6
Brasil (66)	81,8	27,3	21,2	6,1	1,5	0,0	6,1
Argentina (57)	75,4	10,5	5,3	3,5	1,8	8,8	22,8
Colômbia (41)	61,0	51,2	2,4	29,3	31,7	31,7	0,0
Chile (37)	83,8	13,5	8,1	8,1	5,4	0,0	0,0
Peru (22)	77,3	13,6	13,6	18,2	9,1	4,5	0,0
Cuba (22)	77,3	18,2	0,0	31,8	22,7	54,5	0,0

Cruzar editor com país revela que a aparente uniformidade do VS Code esconde perfis culturais bem distintos (Tabela 3). O **Brasil** apresenta um perfil singular: a maior adoção de Vim da Final (21,2%), combinada com nenhum uso de Code::Blocks (0%) e o menor uso de PyCharm (1,5%), exatamente o inverso do perfil de Cuba e Bolívia. Estas duas se inclinam ao **Code::Blocks** (54,5% e 42,1%). **Colômbia** é a mais "Intellij" (29,3% CLion + 31,7% PyCharm). A **Argentina** mantém um nicho próprio de Geany (22,8%), incomum em outros países. **México** e **Chile** concentram-se mais em VS Code (82% e 84%).

Os perfis se acentuam no PDA, onde sobem ao palco os melhores de cada país. No PDA 2026 (12 equipes brasileiras), **50% das equipes do Brasil usaram Vim**, contra 0% no México (6 equipes) e 0% na Bolívia (43º colocado, 1 equipe). A Argentina manteve o nicho de Geany (28,6% das suas 7 equipes). É a evidência mais clara de cultura local de ferramentas atuando, num campo onde o *hardware* é o mesmo para todos.

5. Editores estagnados e ausências surpreendentes

Olhar a adoção pelo prisma da maturidade do projeto traz uma surpresa. O **Code::Blocks** [8] ficou cinco anos sem lançamento estável, da versão 20.03 (março de 2020) à 25.03 (maio de 2025); como nossa imagem é baseada no Ubuntu 24.04, as máquinas ainda trazem a 20.03, então os times competiram com um editor já bem datado. Ainda assim aparece em 11% a 21% das equipes nas fases brasileiras (Regional e Final) e atinge **54,5% em Cuba e 42,1% na Bolívia** na Final. A leitura mais plausível é cultural: disciplinas introdutórias de C/C++ formaram gerações inteiras em Code::Blocks, e o hábito sobrevive a longos hiatos do projeto. Já o Geany [9], comumente colocado no

mesmo balaio de "editores antigos", segue em manutenção ativa, com a versão 2.0 em 2023 e a 2.1 em 2025; o nicho argentino de 22,8% na Final é coerente com leveza somada a desenvolvimento corrente.

Pelo lado oposto chama atenção o **Emacs** [4]: projeto muito ativo (a versão 30 saiu em 2025) e mesmo a 29 do Ubuntu 24.04 já traz *tree-sitter* e cliente LSP, mas sua adoção nunca passou de 5% (0,2% a 4,5%). Falta-lhe o que faz do VS Code o vencedor: setup mínimo, *debugger* gráfico embutido e extensões pré-empacotadas para C++ competitivo; configurar o Emacs a esse nível exige tempo que poucos times investem.

6. Considerações finais

A leitura central é de associação, não de causa: o caminho mais plausível é o inverso, e os editores leves aparecem em cerca de uma de cada quatro equipes do Top-30 nos seis eventos. **Para treinadores**, a escolha modal de quem chega ao topo ainda é o VS Code, mas vale incluir um editor leve (Vim, gedit ou Geany) no repertório de treino. **Para times brasileiros**, o perfil "Vim no topo" é uma marca local: 21% na Final e 50% no PDA.

Um ponto a acompanhar é o **CLion**, cuja adoção subiu entre os ciclos (8,8% para 12,5% na Regional e 12,1% para 16,4% na Final), na contramão do "vale dos IDEs". O movimento coincide com a chegada do *Full Line Code Completion*, a complementação de código por IA local embarcada nas IDEs da JetBrains [11]. Fica a pergunta para os próximos ciclos: a complementação por IA atrairá mais times ao CLion, ou, numa prova cronometrada, atrapalha mais do que ajuda? A própria adoção responderá.

Referências

[1] Davi Antônio da Silva Santos and Bruno César Ribas. Maratona linux a tale of upgrading from ubuntu 20.04 to 22.04. Disponível em: <https://arxiv.org/abs/2510.15263>, 2025.

[2] Cassio P de Campos and Carlos E Ferreira. Boca: um sistema de apoio a competições de programação. In *Workshop de Educação em Computação*. Sociedade Brasileira de Computação, 2004. Disponível em: <https://research.tue.nl/files/134832343/decampos2004d.pdf>. Acesso em: maio 2026.

[3] Geany Project. Geany. Disponível em: <https://www.geany.org>. Acesso em: maio 2026.

[4] GNU Project. Gnu emacs. Disponível em: <https://www.gnu.org/software/emacs/>. Acesso em: maio 2026.

[5] JetBrains. IntelliJ idea, clion e pycharm. Disponível em: <https://www.jetbrains.com>. Acesso em: maio 2026.

[6] Microsoft. Visual studio code. Disponível em: <https://code.visualstudio.com>. Acesso em: maio 2026.

[7] Wall Berg Moraes and Bruno Cesar Ribas. Maratona-linux: um ambiente para a maratona de programação. *Anais do Computer on the Beach*, pages 416–426, 2019. Disponível em: https://www.researchgate.net/publication/332247662_Maratona-Linux_um_ambiente_para_a_Maratona_de_Programacao. Acesso em: maio 2026.

[8] Anton Semenkin et al. Full line code completion: Bringing AI to desktop. Disponível em: <https://arxiv.org/abs/2405.08704>, 2024. Acesso em: maio 2026.

[9] The Code::Blocks Team. Code::blocks. Disponível em: <https://www.codeblocks.org>. Acesso em: maio 2026.

[10] The GNOME Project. gedit. Disponível em: <https://apps.gnome.org/Gedit/>. Acesso em: maio 2026.

[11] Vim Project. Vim — the ubiquitous text editor. Disponível em: <https://www.vim.org>. Acesso em: maio 2026.