Mac OS X 版

Eclipse の導入と プログラムの作成方法



このドキュメントは下記のシステムで検証しました。

目 次

Α.	Ec	lipse を日本語化する
	1.	ダウンロードと解凍
	2.	features フォルダ内のファイルをコピーする
	3.	plugins 内のファイルをコピーする4
В.	Ec	lipse の起動と設定
	1.	起動
	2.	ワークスペースの指定
	3.	「ようこそ」画面・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・
	4.	使いやすい画面にする・・・・・・7
	5.	設定ファイルのインポート・・・・・.7
	6.	C99 プロジェクトファイルをインポートする
	Ľ	注意】C99. zip がインポートできない・・・・・・・・・・・・・・・・・・・・・・・13
C.	プ	ログラムを作成して動作を確認する
	1.	プロジェクトを作る
	2.	プログラムを作成する・・・・・・16
	3.	プログラムの実行 ・・・・・・19
D.	章	が変わったときの操作・・・・・・20
Ε.	設	定ファイルを自分で作成する方法
	1.	環境設定を起動する・・・・・・21
	2.	行番号設定 · · · · · · · · · · · · · · · · · · ·
	3.	斜体字を使わない設定 ······22
	4.	コードテンプレート(simple template)の作成
	5.	設定ファイル (conf.epf)をエクスポートする
F.	C9	9. zip ファイルを自分で作成する方法
	1.	C99 プロジェクトを作成する·····28
	2.	プロジェクトのプロパティを設定する

-2-

A. Eclipse を日本語化する

1. ダウンロードと解凍

http://powercampus.jp/top/helloC/indexMAC.html の説明に従って、必要なファイルをダウ ンロードします。次は、個人フォルダの中の Applications フォルダにダウンロードしています。 先に eclipse-cpp・・・. tar.gz を解凍し、次に NLpackja-・・・. zip を解凍します。すると、この図 のように eclipse と eclipse2 の二つのフォルダができます。

ind	er ファイル 編集 表	示 移動 ウインドウ	ヘルフ	7
0	0 🙆			Applications
C			•	*-
	デバイス	Adobe Photoshop CS	5.1 ト	🖹 c99.zip
	Macintosh HD	Applications	►	eclipse
	Disk iDisk	💿 サイト	Þ	eclipse 2
		💿 ダウンロード	Þ	eclipse-cpp-indigo-SR1-incubation-macosx-cocoa.tar.gz
	場所	📄 デスクトップ	Þ	myconfCpp.epf
	M デスクトップ	💿 パブリック	Þ	NLpackja-eclipse-cpp-galileo-SR1-blancofw.zip
	🆀 morishitayouhei	回 ピクチャ	- P	
	🕂 アプリケーション	🔟 ミュージック	Þ	
	音類	🛅 ムービー	Þ	
_	40-10T	📠 ライブラリ	Þ	
		🛅 書類	Þ	
	0 70			
	□ 昨日			

2. features フォルダ内のファイルをコピーする

図のように eclipse2/features/ を開いて、内のすべてのファイルを選択し[コピー]を選びま す。これは NLpack の方のフォルダです。



■ features Thotoshop CS5.1 ons eclipse 2 - ド y7 y7 y7 y7 y 7 y 7	ウインドウ	ヘルフ	Ĵ	A1 0	0 * 6	• 🕢 🖪 x 2	2:06
■ ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●			🚞 features		<u> </u>		
hotoshop CS5.1 ト C99.zip ons ト eclipse 2 - ド eclipse-cpp-indigo-SR1-incubation-macosx-cocoa.tar.gz ップ ト NLpackja-eclipse-cpp-galileo-SR1-blancofw.zip リ ト ト ト Eclipse app リ ト ト ト ト Eclipse app リ ト ト ト ト ト Eclipse app ロ notice.html ロ 2 ロ 1 2 1 2 1 2 2 2 3 2 3 2 3 2 3 2 3 2 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3		•	*-		Q		
	hotoshop CS5 ions ード ップ ク ック		 C99.zip eclipse eclipse 2 P eclipse-cpp-indigo-SR1-incubation-macosx-cocoa.tar.gz myconfCpp.epf NLpackja-eclipse-cpp-galileo-SR1-blancofw.zip 	artifacts.xml configuration dropins eclipse Eclipse.app epi-v10.html features notice.html p2 plugins readme	開く ゴミ箱に2 情報を見る "features" 複製 エイリア2 "features" 項目をペー 表示オプシ ラペル: ×	 org.eclipse091 org.eclipse091 org.eclipse091 org.eclipse091 org.eclipse091 org.eclipse091 org.eclipse106 var.eclipse106 var.eclipse091 org.eclipse091 org.eclipse09	51620 51620 51620 51620 08.jar 620 620 706 620 706 620 706 620 ,jar 620 620 ,jar 620 620 ,jar 620 620 ,jar 620 620 ,jar 620 620 ,jar 620 620 ,jar 620 620 ,jar ,jar ,jar ,jаг ,jаг ,јар ,јар ,јар ,јар ,јар ,јар ,јар ,јар

次に、eclipse/features/ にすべてのファイルを貼り付けます。

3. pl ugi ns 内のファイルをコピーする

図のように ecl i pse2/pl ugi ns/を開いて、内のすべてのファイルを選択し「コピー」を選びます。



次に、eclipse/plugins/にすべてのファイルを貼り付けます。



B. Eclipse の起動と設定

1. 起動

eclipse フォルダの Eclipse. app を起動します。

ワインドウ	ヘル	プ				M1 🛛 🕙	*	((;-	• 🗠 A
			🚞 eclipse						
	0) 🔅	•				٩		
otoshop CS5	.1 ⊳	Th.	c99.zip			artifacts.xml			▼プレピュー:
ons	►		eclipse	►		configuration		- P-	
	Þ		eclipse 2	⊳		dropins		⊳	
- ۲	- Þ	P	eclipse-cpp-indigo-SR1-incubation-macosx-cocoa.tar.gz		,	eclipse			1000
ップ	- Þ		myconfCpp.epf		0	Eclipse.app			
7	- Þ	P	NLpackja-eclipse-cpp-galileo-SR1-blancofw.zip		•	🕴 epl-v10.html 💺			
	- Þ					features		- P-	
ック	- Þ				•	notice.html			
	- Þ					p2		Þ	
J	- Þ					plugins		- P-	名前
	Þ					readme		⊳	種類
									サイズ
									作成日
									変更日
	_								最後に開いた日

2. ワークスペースの指定

ソースコードや実行可能プログラムを保存するフォルダをワークスペースといいます。C 言語では、 各章ごとにワークスペースを作成します。

Eclipse を起動するとワークスペースの場所を聞いてくるので、ここでは第1章用として、文書フォルダ内に chap01 という名前で作成するよう指定しています。



3. 「ようこそ」画面

右図のような「**ようこそ**」という画面が表 示されますが、×印をクリックして消してく ださい。



-6-

Copyright (C) Takashi Kawaba 2012

4. 使いやすい画面にする

左端のプロジェクト・エクスプローラーだけを残して、後のサブウィンドウを消してください。

C/C++ - Eclipse - /Users/morishitayouhei/Documents/chap01 C <th>Eclipse</th> <th>ファイル(F)</th> <th>編集(E)</th> <th>ソース(S)</th> <th>リファクタリング</th> <th>(T)</th> <th>ナビゲート(N)</th> <th>検索(A)</th> <th>実行(R)</th> <th>プロジェ</th> <th>クト(P)</th> <th>ウイ</th> <th>ンドウ</th>	Eclipse	ファイル(F)	編集(E)	ソース(S)	リファクタリング	(T)	ナビゲート(N)	検索(A)	実行(R)	プロジェ	クト(P)	ウイ	ンドウ
	000	<u></u>		C/C++ - E	clipse - /Users/mo	rishit	ayouhei/Docun	nents/chap	p01				
] 🗗 • 🗐 🔞] 釣 • 仰 • 역	⊴] <mark>:@•63•</mark> ⊙•⊙•	c •••]	% •⊛•]:	\$• 0 • 4 •] @	60	<i>*</i> •] 🗉 🖬			E	9 🗟 c/c	++	
	10ジェクト	・エクスプロ 8	3 - 6	ו					- 0	• M 8	"2		- 0
		Ē	<u>a</u> 4 5 <	7									
0 相目 説明 ▲ リソース バス ロケーション 型				🔛 問題 🕿	3 929 🗉 🗆	ンソー	ル 🔲 プロパティ・	-)				5	
				0項目					1				
				記明			A .	リソース	7.2		ロケーショ	עו	型
] 0°													24.5
	1.0*							1					1414

5. 設定ファイルのインポート

あらかじめ作成してある設定ファイル(myconfCpp. epf)をインポートします。 [ファイル] → [インポート] と選択してください。



[インポート]ダイアログが開きます。[一般]→[設定]と選択してください。

	インボート	
観択 ローカル・ファイル・	システムから設定をインボートします。	Ľ
インボート・ソースの	選択(S):	
フィルター入力		
▼ 🗁 一般		
🗓 アーカイブ	・ファイル	
()ファイル・:	システム	
「一既存プロジ	ェクトをワークスペースへ	
📕 設定		
▶ 🧁 C/C++		
CVS		
► 🤁 Git		
P - Install		
> a g z h	15	
► C= F-4		
▶ → 実行/デバッグ		
?	< 戻る(B) 次へ(N) > キャンセ	ル 完了(F)
<u> </u>		

[設定のインポート]ダイアログが開きます。[参照]をクリックしてください。

000	設定のインポート	
設定のインボート 20 設定ファイルが存在し	ないか、ディレクトリーです。	
ソース設定ファイル(P):	(● ● 願(R)
マイルター入力		
説明(T):		
(すべて選択(5))	(選択をすべて解除(D))	
(?)	(<戻る(B)) 次へ(N) > (キャン)	マセル (売7(F))

▲ ▶ (;; ≡ □)	Applications 🛟 Q	
	名前	▲ 変更日
	DS Store	今日, 22:05
Macintosh HD	C99.zip	今日、21:02
Disk	eclipse	今日、22:10
場所	eclipse 2	09/12/16
100 デスクトップ	eclipse-cpp-indicosx-cocoa.tar.gz	今日、20:54
🏠 morishitayouhei	myconfCpp.epf	今日、21:02
A アプリケーション	NLpackja-eclipseSR1-blancofw.zip	今日、20:54
(3) 書類		
检查		
() 今日		
 昨日 		
④ 過去1週間		
	*.epf	
	(キャンセル)	開く
(新規フォルダ)		

ダウンロードしておいた myconfCpp. epf を選択して [開く] をクリックします。

[完了]をクリックします。これで設定は終了です。

000	設定のインポート	
設定のイン ローカル・	ボート ファイル・システムから設定をインポートします。	
ソース設定	ファイル(P): //Users/morishitayouhei/Applications/myconfC・) (参照(R)
🗹 রুশবের	'ンポート(A)	
74	レター入力	
	C/C++ Editor Appearance C/C++ Editor Behavior	
脱明(T		
(4)	<て選択(5) 選択をすべて解除(D)	
?	< 戻る(B) 次へ(N) > (キャンセル)	完了(7)

6. C99 プロジェクトファイルをインポートする

簡単にプロジェクトを作れるように、ひな形となる C99. zip をインポートします。 [ファイル] → [インポート] と選択してください。



[インポート] ダイアログが開きます。

[既存プロジェクトをワークスペースへ]を選択してください。



-10-

zip形式のファイルをインポートするので [アーカイブ・ファイルの選択] をチェックしてください。

● ○ ● インボート	
プロジェクトのインボート 既存の Eclipse プロジェクトを検索するディレクトリーを選択します。	
 ○ ルート・ディレクトリーの選択(T): ○ スーカイゴ・ファイルの選択(A): 	参照(R)
(A):	\$\$99.R(R)
7 □ 9 ± 9 ト(P):	すべて選択(S) (選択をすべて解除(D)) リフレッシュ(E)
✓ プロジェクトをワークスペースにコピー(C)	
ワーキング・セット	
□ ワーキング・セットにプロジェクトを追加(T)	
ワーキング・セット(0):	\$ 選択(E)
? < 戻る(B) 次へ(N) > (キャン)	1セル 完了(F)

ファイルを選択するために [参照] をクリックします。

	インボート	
プロジェクトのインボート 既存の Eclipse プロジェク	・ トを検索するディレクトリーを選択します。	
○ ルート・ディレクトリー	の選択(T):	《参照(R)
アーカイブ・ファイルの)選択(A):	参照(R)
ブロジェクト(P):		
		(
		選択をすべて解除(D)
		(リフレッシュ(E)
✓ プロジェクトをワーク:	スペースにコピー(C)	
 ✓ プロジェクトをワーク: ワーキング・セット 	スペースにコピー(C)	
 ✓ プロジェクトをワーク: ワーキング・セット □ ワーキング・セット 	スペースにコピー(C) : プロジェクトを追加(T)	
 ✓ フロジェクトをワーク: ワーキング・セット □ ワーキング・セットは ワーキング・セット(0): 	スペースにコピー(C) :プロジェクトを追加(T)	\$〕 (選択(E))
 ✓ プロジェクトをワーク: ワーキング・セット □ ワーキング・セット □ ワーキング・セット(0): 	スペースにコピー(C) - プロジェクトを追加(T)	\$〕 (選択(E))
 ✓ プロジェクトをワーク: ワーキング・セット □ ワーキング・セットは ワーキング・セット(0): 	スペースにコピー(C) :プロジェクトを追加(T)	: 選択(E)
 ✓ プロジェクトをワーク: ワーキング・セット □ ワーキング・セット □ ワーキング・セット(0): 	スペースにコピー(C) :プロジェクトを追加(T)	÷) 選択(E)

ダウンロードしておいた [c99. zip] を選択します。

000	インポート	
	🛛 🔝 Applications 🔹 🔍	
 デバイス Macintosh HD iDisk 潮所 デスクトップ 介 morishitayouhei アプリケーション 書類 検索 今日 昨日 過去1週間 	名前 DS_Store C99.zip C09.zip C99.zip	 変更日 今日、22:05 今日、21:02 今日、22:10 09/12/16 今日、20:54 今日、21:02 今日、20:54
新規フォルダ	(*.jar;*.zip;*.tar;*.tar.gz;*.tgz ・)	

[完了]をクリックして、インポートします。

● ○ ○ インポート	
プロジェクトのインポート 既存の Eclipse プロジェクトを検索するディレクトリーを選択します。	
 ルート・ディレクトリーの選択(T): アーカイブ・ファイルの選択(A): morishitayouhei/Applications/c99 プロジェクト(P): 	参照(R) .zip 参照(R)
✓ c99 (c99)	すべて選択(S) 選択をすべて解除(D) リフレッシュ(E)
 ✓ ブロジェクトをワークスペースにコピー(C) ワーキング・セット ワーキング・セットにプロジェクトを追加(T) ワーキング・セット(O): 	送 選択(E)
(N) > キャンセル	· 余7(F)

以上で、次の図のように **c99** というひな形のプロジェクトが作成されます。 これ以降、プロジェクトを作成する時は、これをコピーします。

Eclipse	ファイル(F)	編集(E)	ソース(S)	リファクタリング(T)	ナビゲート(N)	検索(A)	実行(R)
00			C/C-	++ - Eclipse - /Users/m	norishitayouhei/D	ocuments/	chap01
] 📬 - 🔚 🛛	d 🕒 🔒 🛛 💰)• 🚳 • 🖻	• @•] </th <th>· ⊚ •] 救• Q• Q•] (</th> <th>] • 🖉 😂 🥶</th> <th>1</th> <th>• 🖓 • 🐤 🤇</th>	· ⊚ •] 救• Q• Q•] (] • 🖉 😂 🥶	1	• 🖓 • 🐤 🤇
トレン プロジェ	クト・エクスプロ	23					
		8 🕸 👔	, ▼				
▶ 😂 c99							
he i							
· ·							
] 🗣	0 項目が選択され	ました]

【注意】C99. zip がインポートできない

C99 をインポートする前に、何らかの事情で自分で C99 プロジェクトを作成した場合、それを削除して C99. zip をインポートしようとしても、うまくインポートできない場合があります。

それは、プロジェクトを削除するときに、次のように、[ディスク上からプロジェクト・コンテンツ を削除] にチェックを入れないかぎり、Eclipse 上では見えなくなっても、フォルダ自体は消えずに 残ってしまうからです。



-13-

この場合、インポート処理で次のように表示されます。

● ○ ◎ インポート
プロジェクトのインポート いくつかのプロジェクトは、ワークスペース・ディレクトリーにすでに存在するた め、インポートできません
○ ルート・ディレクトリーの選択(T):
● アーカイブ・ファイルの選択(A): /Users/morishitayouhei/Applications/ 参照(R)
プロジェクト(P):
c99 (c99) すべて選択(S) 選択をすべて解除(D) リフレッシュ(E)
✓ プロジェクトをワークスペースにコピー(C)
ワーキング・セット
□ ワーキング・セットにプロジェクトを追加(T)
ワーキング・セット(0): 選択(E) 選択(E)
(?) (天る(B)) 次へ(N) > キャンセル 完了(F)

このように表示された場合、Finder でワークスペースの中を見て、C99 フォルダを削除する必要が あります。例えば、ワークスペースを [文書] フォルダの中に [chap01] という名前で作ったのであ れば、chap01 フォルダ内にある [C99] フォルダを削除してください。

00		C99		0	
		11.33 GB 空き			
Adobe Photoshop CS5.1	Þ	📨 2012年年賀状.jpg		🚞 c99 🔹 🕨	
Applications	⊳	20110122_0059.JPG		K	
🔯 サイト	Þ	🚞 chap01	•		
🔯 ダウンロード	Þ	photoshop	⊳		
📄 デスクトップ	Þ	📄 prac	Þ		
💿 パブリック	Þ	🖹 スタックについて.pdf			\mathbf{k}
🛅 ピクチャ	Þ				> ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
🔟 ミュージック	Þ				
🛅 ムービー	Þ				当時会子フ
📠 ライブラリ	Þ				削床 9 つ
◎ 書類	►				
(-				

-14-

C. プログラムを作成して動作を確認する

1. プロジェクトを作る

プロジェクトは、ソースコードや実行プログラムを入れるフォルダです。ひとつのプログラムにひと つのプロジェクトが必要です。

そこで、ひな形の **c99** プロジェクトをコピーして作成します。**c99** プロジェクトをクリックし、[編 集] メニューで [**コピー**] を選びます。



[編集] メニューで [貼り付け] を選択します。

Eclipse ファイル(F)	編集(E)	ソース(S)	リファクタ
00	<⇒元1	こ戻す(U)	жZ
] 📬 • 🖾 🖻 🙆 🗋 🙋	ゆや!	つ直し(R)	ΰжΖ
10ジェクト・エクスプロ	*切!)取り(T)	жx
	D I E	∠_(C)	жc
▶ 😂 c99	11 11 11 11 11 11 11 11 11 11 11 11 11	O付け(P)	жv
	<mark>×</mark> 削開 すべて	余(D) 選択(A)	æ ⊮A
	検索/i	置換(F)	₩F
	ブック タスク	マークの追加 の追加(S)	П(K)

[プロジェクトのコピー] ダイアログが表示されるので、プロジェクト名を入力して [OK] をクリック します。ここではプロジェクト名として"test"を指定しています。

000	プロジェクトのコピー
プロジェクト名(P): test
🗹 デフォルト・1	コケーションの使用(D)
ロケーション(L):	/Users/morishitayouhei/Documents/chap01/te (参照(R))
	ファイル・システムを選択: デフォルト 🛟
?	(++>+2h) (OK)

以上で次のように test プロジェクトが作成されます。



2. プログラムを作成する

次に、test プロジェクト内にプログラム test. c を作成してみましょう。 図のように、ソースファイル作成ボタンをクリックしてください。



[新規ソース・ファイル] ダイアログが表示されます。

ソース・ファイル名(プログラム名)として、test.cと入力します。そして、さらにテンプレートを 選ぶために、[テンプレート]をクリックし、simple templateを選択してください。



次の図のようになるので、[完了]をクリックします。

000	新規ソース・ファ	イル	
新規ソース・ファイル 新しいソース・ファイル	を作成します。		C
ソース・フォルダー(D):	test		
ソース・ファイル(E):	test.c		
テンプレート(T):	simple template		
?		(++>+	完了(F)

骨格が出来上がった次のようなソースプログラムが作成され、表示されます。これに必要な部分を書き 足すことでプログラムが完成します。



ここでは、C99の機能を含む次のようなテストプログラムを書き込んでください。 書き終わったら、左上にある保管ボタンをクリックします。

Eclipse	ファイル(F)	編集(E)	ソース(S)	リファク	タリング(T)	ナビゲート(N)
00	_		c C/C++ -	test/test.c	- Eclipse - /	Users/morishita
] ➡ ₽ ₽] ∯ ∲*	È (• 🚳 • 💼	• @•] %•	⊗ •]‡⊳	• • • •	2 🖻 🖨 🖋
7ロジェ	クト・エクスプロ	1 ¹	- 🗆 💽 *te	st.c 🖾		
▶ 2 c99 ▼ 2 test ▶ 2 fest ▶ 2 test	ンクルード st.c	E \$	▼ 1# 2 i 3 4 5 6 7 }	include nt main(vo for(int pri } return	<pre><stdio.h> id){ i=0; i<5; ntf("%d ", 0;</stdio.h></pre>	i++){ i);

3. プログラムの実行

ビルドボタンをクリックして、実行形式のプログラムを作成します。



ビルドすると [**バイナリー**] と [**Debug**] という実行形式のファイルができます。 実行ボタンをクリックすると図のように、コンソールウィンドウに実行結果が表示されます。



D. 章が変わったときの操作

新しい章を学習する際には、ワークスペースも新しくした方がよいでしょう。そうしないと、沢山の プロジェクトができてしまい、使いづらくなります。

新しいワークスペースを作成するには、Eclipse 起動時に指定します。

次の図では第2章のために chap02 と指定して、ワークスペースを切り替えています。chap02 という ワークスペースがない場合は Eclipse が自動的に作成してくれます。

		1.	
	The second s		
	ワークスペース・ランチャー		
. 6	ワークスペースの選択		
1	Eclipse は、ワークスペースと呼ばれるフォルダーにプロジェクトを保管します。 このセッションに使用するワークスペース・フォルダーを選択してください。		
	ワークスペース(W): //Users/morishitayouhei/Documents/chap02		
	この選択をデフォルトとして使用し、今後この質問を表示しない(U)		
	(キャンセル)	ОК	-

E. 設定ファイルを自分で作成する方法

設定ファイル myconfCpp. epf は、自分で作成することができます。ここでは、その作成方法を示します。設定ファイルを作成するには、4ページの「4. 使いやすい画面にする」まで実行しておきます。

1.環境設定を起動する

メニューで、[Eclipse] → [環境設定] と選択します。



2. 行番号設定

[設定] ダイアログが表示されます。[一般] → [エディター] → [テキスト・エディタ] と選択しま す。さらに、右側欄で[行番号の表示] にチェックをつけます。

00	設定			
(フィルター入力)	テキスト・エディター		\$	
▼一般 Web プラウザー	See 'Colors and Fonts' to configure	the font.		
テキスト・エディター	「元に戻す」ヒストリーのサイズ(U):	200		
ファイルの関連付け	表示されるタブ幅(T):	4		
コンテンツ・タイプ	□ タブでスペースを挿入(1)			
▶セキュリティー ▶ネットワーク接続	☑ 現在行の強調表示(G)			
パースペクティブ	□ 印刷マージンの表示(W)			
▶ワークスペース ▶開始およびシャットダウン	印刷マージン列(M):	80		
▶外観	☑ 行番号の表示(B)			
快楽 比較/パッチ	☑ 范囲インジケーターの表示(R)			
►C/C++ Mvlvn	空白文字の表示(O)			
▶ Remote Systems	☑ スティッキーにする方法を吹き出	しに表示(H)		
▶インストール/更新 ▶チーム	ホパー内でのマウス移動時(V):	遅延リッチ化	•	
▶ヘルプ ▶ 体田マータ・コレクター	✓ テキストのドラッグ・アンド・ドロップを使用可能にする(P)			
▶実行/デバッグ				
	☑ 行の先頭と最後にスマート・キャ!	レット配置(5)		

3. 斜体字を使わない設定

 $[C/C++] \rightarrow [エディター] \rightarrow [コンテンツ・アシスト] \rightarrow [構文の色の指定] と選択します。さらに、$ 右側欄で、[コード] をクリックして、[static フィールド] や [static メソッド呼び出し] など、斜体字にチェックが入っているので、チェックを外しておきます。



4. コードテンプレート(simple template)の作成

[コードスタイル] → **[コード・テンプレート**] と選択します。

さらに、右側欄で、[ファイル] → [C ソース・ファイル] と選択し、[新規] ボタンをクリックします。



[新規テンプレート] ダイアログが開きます。

[名前]欄に"Simple Template"と入力し、さらに、プログラムの骨格となるコードを(空行も含めて)次のように入力します。完成したら [OK] をクリックしてください。

000	新	i規テンプレー	F		
名前(N): 説明(D):	Simple Template	型(T):	C 7-2.771	11	:
ノ ミターン(P) :	<pre>#include <stdio.h> int main(){ return 0; }</stdio.h></pre>				
	変数の挿入(M)				
?			(**>	en)	ок

次のように Simple Template が作成されます。



5. 設定ファイル (conf. epf)をエクスポートする

適当なファイル名を付けて(設定ファイルの名前は任意)エクスポートすると設定ファイルを作成で きます。メニューで、[ファイル]→[エクスポート]と選択すると、[エクスポート]ダイアログが表示 されます。

ダイアログで、[一般]→[設定]と選択します。



何という名前でどのフォルダに出力するか指定します。[参照]ボタンをクリックしてください。

000)	設定のエクスポート	
設定のエ (3) 設定フ	クスポート ファイルが未設フ	宅であるか、通常のファイルではありません。	
₫ す ~7	エクスポート(A)	
	C/C++ Edit C/C++ Edit	or Appearance or Behavior	
脱明	(T):		
3	すべて選択(S)	(選択をすべて解除(D))	
宛先設定	ファイル(P): [● (参照(R))
□ 警告	を出さずに既存	ファイルを上書き(0)	
?		(<戻る(B) 次へ(N) > (キャ)	シセル 売了(月)

ファイルダイアログが表示されます。名前には、"conf.epf"など、適当な名前を指定し、保存する フォルダを選択してから [保存] ボタンをクリックします。フォルダはどこでも構いませんが、この例 では、文書フォルダを指定しています。

	Applications	
▼ デバイス 通 Macintosh HD ■ iDisk ▼ 場所 ■ デスクトップ 全 morishitayouhei → アフリケーション	名前 DS_Store C99.zlp C99.zl	▲ 変更日 今日、22:05 今日、21:02 今日、22:48 09/12/16 ○ 今日、20:54 今日、21:02 ○ 今日、20:54
[*] 會現 ▼ 拾蜜		

元の [設定のエクスポート] ダイアログに戻るので、「完了」をクリックすると終了です。

00	設定のエクスポート	
設定のエクスポート 設定をローカル・フ	・イル・システムにエクスポートします。	
	~(A)	_
C/C++ C/C++	litor Appearance litor Behavior	٦
説明(T):		
(すべて選択)) (選択をすべて解除(D))	
宛先設定ファイル(/Users/morishitayouhei/Applications/conf.epf 💽 参照(R)	
□ 警告を出さずに	存ファイルを上書き(0)	
?	<戻る(B) 次へ(N) > キャンセル 完了(F)	

F. C99. zip ファイルを自分で作成する方法

プロジェクトのひな形ファイル C99. zip も自分で作成できます。

1. C99 プロジェクトを作成する

メニューで、**[ファイル]** → **[新規]** → **[C プロジェクト**] と選択します。

[C Project] ダイアログが表示されるので、下図のように、"C99"という名前を入力し、[実行可 能] な[Empty Project]を選択し、さらに [MacOSX GCC] を選択して、[完了] をクリックしてく ださい。



2. プロジェクトのプロパティを設定する

メニューで [**プロジェクト**] → [**プロパティ**] と選択します。

🗯 Eclipse 🗇	ファイル(F)	編集(E)	ソース(S)	リファクタリング(T)	ナビゲート(N)	検索(A)	実行(R)	プロジェクト(P)	ウィンドウ(W)	ヘルプ(H
00			C/C++	- Eclipse - /Users/mo	rishitayouhei/Doo	uments/c	hap01	プロジェクトを	開<(E)	
] 📬 - 🔛 🕼 📤	🗟] 💣 • (📬 • 💽 • 🖸	3 •] 🗞 • 🛞	•] 🏇 • 🜔 • 🏊 •] 🍅	🖉 🖨 🖉 •] 🔲	n]2-1	3 • % (-	プロジェクトを	闭じる(5)	
🎦 プロジェクト・コ	ウスプロ 🔀	- 6	ו						≺(A)	жв
	E	4 5 <	7					構成のビルト プロジェクトの	ビルド(B)	4
🕨 🚔 C99								ワーキング・セ	ットのビルド(W)	►
								クリーン(N) ✓ 自動的にビルド	(M)	
								Make A-fw	ь. Ь	
								Make 9 99	1-	-
								フロパティー(P)	

[**プロパティ**] ダイアログが開きます。

ダイアログでは、左側欄で、[C/C++ビルド]→[設定]と選択します。

00		プロパティー: C99			
(フィルター入力) ▶リソース	設定				
+ C/C++ E/DF Logging ツール・チェーン・エディ ディスカバリー・オプショ ビルド変数 環境 設定 ▶ C/C++ 一般	Configuration: Debug [Active	・1 ・1 Command:	ステップ gcc	🙅 ビルド成果物	📄 パイナリー
▶タスク・リポジトリー ビルダー プロジェクト参照 実行/デバッグ設定	 ライブラリー 没その他 没共用ライブラリー設定 マ い GCC Assembler 	All options:			

さらに、右側欄で、[Cross GCC Compiler] をクリックしてください。



コンパイラが、最新の C99 のモードでコンパイルを行うようにするため、[Command] 欄に指定を書 き足します。"gcc"の部分を、"gcc -std=c99"と書き変えてください。これが C99 でコンパイル するための指定です。

00	プロ	コパティー: C99
(フィルター入力)	設定	Q=• Q • •
▼5/5×××××××××××××××××××××××××××××××××××	Configuration: Debug [Active	1
ビルド変数 環境 設定 ▶C/C++ 一般	⁽¹⁾ MacOS X C Linker	♪ ビルド・ステップ ──ビルド成果物 □→パイナリー・パーサー ⊗ エラー・ Command: gcc -std=c99
WikiText ▶ タスク・リポジトリー ビルダー プロジェクト参照 実行/デバッグ設定	(29一般 29 ライブラリー 29 その他 29 共用ライブラリー設定 ▼ 30 GCC Assembler 29 一般	All options: -00 -g3 -Wall -c -tmessage-length=0
	 CCC C Compiler プリプロセッサー シンボル ディレクトリー 最適化 デバッグ 警告 その他 	Expert settings: Command line pattern: S{COMMAND} S{FLAGS} S{OUTPUT_FLAG} S{OUTPUT_PREFIX}S{OUT
?	·	***>***

次に、[バイナリーパーサー]をクリックします。

000	ブ	コパティー: C	99
(フィルター入力)	設定		0 • 0 • •
 ▶リソース ▼C/C++ ビルド Logging ツール・チェーン・エディ ディスカバリー・オブショ ビルド変数 現境 設定 ▶C/C++ 一般 WikiText ▶タスク・リポジトリー ビルダー プロジェクト参照 実行/デバッグ設定 	Configuration: Debug (Active) PILK - 2	テップ ●ビルド成果物 □ バイナリー・パーサー ● エラー・
	 MacOS X C Linker 一般 ライブラリー ※その他 ※共用ライブラリー設定 ※時 GCC Assembler 	Command: All options:	gcc -std=c99 -O0 -g3 -Wall -c -fmessage-length=0
	 ▼ ● GCC Assembler ○ 一般 ▼ ※ GCC C compiler ○ フリプロセッサー ○ シンポル ○ ディレクトリー ○ 要適化 ○ デバッグ 	Expert settin Command line pattern:	gs: \${COMMAND} \${FLAGS} \${OUTPUT_FLAG} \${OUTPUT_PREFIX}\${OUT

バイナリーバーサーの選択画面になるので、

[Mach-0 64 Parser] と [Mac-0 パーサー] の両方にチェックを入れます。

000	プロパティー: C99	
(フィルター入力)	設定	
 ▶リソース ▼C/C++ ビルド Logging ツール・チェーン・エディ ディスカバリー・オブショ ビルド変数 環境 ▶C/C++ 一般 WikiText ▶タスク・リポジトリー ビルダー プロジェクト参照 実行/デバッグ設定 	③ ツール設定 参 ビルド・ステップ 型 ビルド成果 Binary parser: Mach-O 64 Parser Elf パーサー Elf パーサー Up w Set ビーサー GNU Elf パーサー Cygwin PE パーサー Cygwin PE パーサー AIX XCOFF32 パーサー	
	パイナリー・パーサー・オブション	
	c++filt コマンド:	
	c++filt	参照(B))
(T () () ()
(?)		OK (**>tu)

以上でプロジェクトのプロパティ設定は終了です。右下にある [OK] ボタンをクリックして終了しま す。また、ここで、Eclipse 自体も、一度、終了してください。

3. zip 圧縮ファイルを作る

ワークスペースのあるフォルダを Finder で見ます。

この例では、ワークスペースは[書類]フォルダの中にある chap01 フォルダです。そこで、この chap01 フォルダの中を見ると、C99 というフォルダがあることが分かります。

Eclipseのプロジェクトは実際にはフォルダなのです。

このフォルダを Zi p 圧縮したものが C99. zi p ファイルです。

Finder のメニューから、[ファイル] → [C99 を圧縮] を選んでください。

inder	ファイル 編集 表示 移動	ウインドウ	ヘルプ		M1 🛛 🕙 🕏
	新規 Finder ウインドウ	жN	i \varTheta 🔿 🔿	C99	
	新規フォルダ	☆ 器N		11.33 GB 空き	
	新規スマートフォルダ	∩⊂ ≋ N	Adobe Photoshop CS5.1 🕨	📨 2012年年賀状.jpg	🚞 C99 🕨 🕨
	新規ディスク作成フォルダ		Applications	20110122_0059.JPG	
1.1.1	このウインドウで開く	^#O	<u> </u>	chap01	•
	このアプリケーションで開く	►	◎ ダウンロード ▶	photoshop	
•	プリシト		■ デスジトップ ト	prac	P
	ワインドワを閉じる	жw			
1	概要情報を見る	~第1	□ ミュージック ト		
			i iii 4-ビ- ►		
	"C99"を圧縮		🛅 ライブラリ 🛛 🕨		
	複製	жD	● 書類 ▶		
	エイリアスを作成	жL			
•	"C99"をクイックルック	жY			
	オリジナルを表示	₩R			
	よく使う項目に追加	^%T		н	
	ゴミ箔に入れる	99 / VI			
	コミ権に入れる 取り出す	~ ₩E	and all the second		

すると、次のように、C99. zipができます。

このファイルをどこか適当な場所に移動しておき、プロジェクトを作成するとき参照できるようにして ください。

